Handbuch

AUTOMATION

WAGO-I/O-SYSTEM 750

ETHERNET ECO Controller

750-852

SPS - Programmierbarer Feldbuscontroller ETHERNET ECO

Version 1.2.0

WAGO®

© 2015 by WAGO Kontakttechnik GmbH & Co. KG
Alle Rechte vorbehalten.

WAGO Kontakttechnik GmbH & Co. KG

Hansastraße 27
D-32423 Minden

Tel.: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: http://www.wago.com

Technischer Support

Tel.: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55

E-Mail: support@wago.com

Es wurden alle erdenklichen Maßnahmen getroffen, um die Richtigkeit und Vollständigkeit der vorliegenden Dokumentation zu gewährleisten. Da sich Fehler, trotz aller Sorgfalt, nie vollständig vermeiden lassen, sind wir für Hinweise und Anregungen jederzeit dankbar.

E-Mail: documentation@wago.com

Wir weisen darauf hin, dass die im Handbuch verwendeten Soft- und Hardwarebezeichnungen und Markennamen der jeweiligen Firmen im Allgemeinen einem Warenzeichenschutz, Markenzeichenschutz oder patentrechtlichem Schutz unterliegen.

Inhaltsverzeichnis

1 Hinweise zu dieser Dokumentation ... 11
1.1 Gültigkeitsbereich ... 11
1.2 Urheberschutz ... 11
1.3 Symbole ... 12
1.4 Darstellung der Zahlensysteme ... 13
1.5 Schriftkonventionen ... 13

2 Wichtige Erläuterungen ... 14
2.1 Rechtliche Grundlagen ... 14
2.1.1 Änderungsvorbehalt ... 14
2.1.2 Personalqualifikation ... 14
2.1.3 Bestimmungsgemäße Verwendung des WAGO-I/O-SYSTEMs 750 14
2.1.4 Technischer Zustand der Geräte ... 15
2.2 Sicherheitshinweise ... 16
2.3 Spezielle Einsatzbestimmungen für ETHERNET-Geräte ... 18

3 Systembeschreibung ... 19
3.1 Fertigungsnummer ... 20
3.2 Hardware-Adresse (MAC-ID) ... 20
3.3 Komponenten-Update ... 21
3.4 Lagerung, Kommissionierung und Transport ... 22
3.5 Aufbaurichtlinien und Normen ... 23
3.6 Spannungsversorgung ... 24
3.6.1 Potentialtrennung ... 24
3.6.2 Systemversorgung ... 25
3.6.2.1 Anschluss ... 25
3.6.2.2 Auslegung ... 26
3.6.3 Feldversorgung ... 28
3.6.3.1 Anschluss ... 28
3.6.3.2 Absicherung ... 30
3.6.4 Versorgungsbeispiel ... 34
3.6.5 Netzgeräte ... 36
3.7 Erdung ... 37
3.7.1 Erdung der Tragschiene ... 37
3.7.1.1 Rahmenaufbau ... 37
3.7.1.2 Isolierter Aufbau ... 37
3.7.2 Funktionserde ... 38
3.8 Schirmung ... 39
3.8.1 Allgemein ... 39
3.8.2 Busleitungen ... 39
3.8.3 Signalleitungen ... 39
3.8.4 WAGO-Schirm-Anschlusssystem ... 40

4 Gerätebeschreibung ... 42
4.1 Ansicht ... 45
4.2 Anschlüsse ... 47
4.2.1 Geräteeinspeisung ... 47
4.2.2 Feldbusanschluss ... 48

4.3 Anzeigeelemente ... 49
4.4 Bedienelemente ... 50
4.4.1 Service-Schnittstelle ... 50
4.4.2 Betriebsartenschalter ... 51
4.4.3 Adresswahlschalter ... 53
4.5 Technische Daten ... 54
4.5.1 Gerätedaten ... 54
4.5.2 Systemdaten ... 54
4.5.3 Versorgung ... 55
4.5.4 Feldbus MODBUS/TCP ... 55
4.5.5 Zubehör ... 55
4.5.6 Anschlusstechnik ... 55
4.5.7 Klimatische Umweltbedingungen ... 56
4.5.8 Mechanische Festigkeit ... 56
4.6 Zulassungen ... 57
4.7 Normen und Richtlinien ... 58

5 Montieren ... 59

5.1 Einbaulage ... 59
5.2 Gesamtaufbau ... 59
5.3 Montage auf Tragschiene ... 61
5.3.1 Tragschieneneigenschaften ... 61
5.3.2 WAGO-Tragschienen ... 62
5.4 Abstände ... 62
5.5 Montagereihenfolge ... 63
5.6 Geräte einfügen und entfernen ... 64
5.6.1 Feldbuskoppler/-controller einfügen ... 65
5.6.2 Feldbuskoppler/-controller entfernen ... 65
5.6.3 Busklemme einfügen ... 66
5.6.4 Busklemme entfernen ... 67

6 Geräte anschließen ... 68

6.1 Datenkontakte/Klemmenbus ... 68
6.2 Leistungskontakte/Feldversorgung ... 69
6.3 Leiter an CAGE CLAMP® anschließen ... 70

7 Funktionsbeschreibung ... 71

7.1 Betriebssystem ... 71
7.1.1 Anlauf des Feldbuscontrollers ... 71
7.1.2 PFC-Zyklus ... 71
7.2 Prozessdatenaufbau ... 73
7.2.1 Prinzipieller Aufbau ... 73
7.2.2 Beispiel für ein Eingangsprozessabbild ... 75
7.2.3 Beispiel für ein Ausgangsprozessabbild ... 76
7.2.4 Prozessdaten MODBUS/TCP und EtherNet/IP ... 77
7.3 Datenaustausch ... 78
7.3.1 Speicherbereiche ... 80
7.3.2 Adressierung ... 83
7.3.2.1 Adressierung der Busklemmen ... 84
7.3.2.2 IEC-61131-3-Adressräume ... 85
7.3.2.3 Absolute Adressierung ... 86

7.3.3 Datenaustausch MODBUS/TCP-Master und Busklemmen ... 88
7.3.4 Datenaustausch EtherNet/IP-Master und Busklemmen ... 90
7.3.5 Datenaustausch SPS-Funktionalität (CPU) und Busklemmen ... 91
7.3.6 Datenaustausch Master und SPS-Funktionalität (CPU) ... 92
7.3.6.1 Beispiel MODBUS/TCP-Master und SPS-Funktionalität (CPU) . 92
7.3.7 Anwendungsbeispiel ... 94

8 In Betrieb nehmen ... 95
8.1 PC und Feldbusknoten anschließen ... 96
8.2 IP-Adresse an den Feldbusknoten vergeben ... 96
8.2.1 IP-Adresse mit dem Adresswahlschalter vergeben ... 97
8.2.2 IP-Adresse mit DHCP vergeben ... 99
8.2.2.1 DHCP aktivieren ... 99
8.2.2.2 IP-Adresse dauerhaft durch Option „use IP from EEPROM“ ... 100
8.2.3 IP-Adresse mit WAGO-ETHERNET-Settings vergeben ... 102
8.2.4 IP-Adresse über ein SPS-Programm vergeben ... 104
8.2.5 IP-Adresse mit einem BootP-Server vergeben ... 104
8.2.5.1 MAC-ID ermitteln ... 105
8.2.5.2 IP-Adresse ermitteln ... 107
8.2.5.3 IP-Adresse vergeben ... 108
8.2.5.4 IP-Adresse dauerhaft durch Option „use IP from EEPROM“ ... 108
8.2.5.5 Gründe für eine fehlgeschlagene IP-Adressvergabe ... 110
8.3 Funktion des Feldbusknotens testen ... 111
8.4 Flash-Dateisystem vorbereiten ... 113
8.5 Uhrzeit und Datum einstellen ... 115
8.6 Werkseinstellungen wiederherstellen ... 118

9 PFC mit WAGO-I/O-*PRO* programmieren ... 119
9.1 Feldbuscontroller mit dem I/O-Konfigurator konfigurieren ... 122
9.1.1 Feldbuscontroller mit der Datei „EA-config.xml“ konfigurieren ... 125
9.2 ETHERNET-Bibliotheken für WAGO-I/O-*PRO* ... 127
9.3 Einschränkungen im Funktionsumfang ... 128
9.4 Generelle Hinweise zu den IEC-Tasks ... 130
9.4.1 Ablaufschema einer IEC-Task ... 132
9.4.2 Die wichtigsten Task-Prioritäten im Überblick ... 132
9.5 Systemereignisse ... 134
9.5.1 Systemereignisse aktivieren/deaktivieren ... 134
9.6 IEC-Programm auf den Controller übertragen ... 136
9.6.1 Applikation mittels serieller Service-Schnittstelle übertragen ... 137
9.6.2 Applikation mittels Feldbus via ETHERNET übertragen ... 140

10 Im Web-Based-Management-System (WBM) konfigurieren ... 142
10.1 Information ... 144
10.2 Ethernet ... 146
10.3 TCP/IP ... 149
10.4 Port ... 151
10.5 Watchdog ... 153
10.6 Clock ... 155
10.7 Security ... 157
10.8 Modbus ... 160
10.9 EtherNet/IP ... 162

10.10 PLC-Info ... 164
10.11 PLC ... 165
10.12 Features ... 167
10.13 I/O Config ... 168
10.14 Disk Info ... 170

11 Diagnose ... 171

11.1 LED-Signalisierung ... 171
11.1.1 Feldbusstatus auswerten ... 171
11.1.2 Knotenstatus auswerten – I/O-LED (Blinkcode-Tabelle) ... 173
11.1.2.1 USR-LED ... 181
11.2 Fehlerverhalten ... 182
11.2.1 Feldbusausfall ... 182
11.2.2 Klemmenbusfehler ... 184

12 Feldbuskommunikation ... 185

12.1 Implementierte Protokolle ... 185
12.1.1 Kommunikationsprotokolle ... 185
12.1.1.1 IP (Internet Protocol) ... 185
12.1.1.2 TCP (Transmission Control Protocol) ... 190
12.1.1.3 UDP (User Datagram Protocol) ... 191
12.1.2 Konfigurations- und Diagnoseprotokolle ... 191
12.1.2.1 BootP (Bootstrap Protocol) ... 191
12.1.2.2 DHCP (Dynamic Host Configuration Protocol) ... 193
12.1.2.3 HTTP (Hypertext Transfer Protocol) ... 195
12.1.2.4 DNS (Domain Name Systems) ... 196
12.1.2.5 SNTP-Client (Simple Network Time Protocol) ... 196
12.1.2.6 FTP-Server (File Transfer Protocol) ... 197
12.1.3 Anwendungsprotokolle ... 198
12.2 MODBUS-Funktionen ... 199
12.2.1 Allgemeines ... 199
12.2.2 Anwendung der MODBUS-Funktionen ... 202
12.2.3 Beschreibung der MODBUS-Funktionen ... 203
12.2.3.1 Funktionscode FC1 (Read Coils) ... 204
12.2.3.2 Funktionscode FC2 (Read Discrete Inputs) ... 206
12.2.3.3 Funktionscode FC3 (Read Holding Registers) ... 208
12.2.3.4 Funktionscode FC4 (Read Input Registers) ... 209
12.2.3.5 Funktionscode FC5 (Write Single Coil) ... 210
12.2.3.6 Funktionscode FC6 (Write Single Register) ... 211
12.2.3.7 Funktionscode FC11 (Get Comm Event Counter) ... 212
12.2.3.8 Funktionscode FC15 (Write Multiple Coils) ... 213
12.2.3.9 Funktionscode FC16 (Write Multiple Registers) ... 215
12.2.3.10 Funktionscode FC22 (Mask Write Register) ... 216
12.2.3.11 Funktionscode FC23 (Read/Write Multiple Registers) ... 217
12.2.4 MODBUS-Register-Mapping ... 219
12.2.5 MODBUS-Register ... 222
12.2.5.1 Zugriff auf Registerwerte ... 223
12.2.5.2 Watchdog-Register ... 223
12.2.5.3 Diagnoseregister ... 229
12.2.5.4 Konfigurationsregister ... 230
12.2.5.5 Firmware-Informationsregister ... 236

12.2.5.6 Konstantenregister ... 238
12.3 EtherNet/IP (Ethernet/Industrial Protocol) ... 240
12.3.1 Allgemeines ... 240
12.3.2 Protokollübersicht im OSI-Modell ... 241
12.3.3 Eigenschaften der EtherNet/IP-Protokollsoftware ... 242
12.3.4 EDS-Datei ... 242
12.3.5 Objektmodell ... 243
12.3.5.1 Allgemeines ... 243
12.3.5.2 Klassen-Übersicht ... 244
12.3.5.3 Erläuterung der Tabellenköpfe in den Objektbeschreibungen ... 245
12.3.5.4 Identity (01 hex) ... 246
12.3.5.5 Message Router (02 hex) ... 248
12.3.5.6 Assembly (04 hex) ... 249
12.3.5.7 Connection (05 hex) ... 254
12.3.5.8 Connection Manager (06 hex) ... 254
12.3.5.9 Port Class Object (F4 hex) ... 254
12.3.5.10 TCP/IP Interface Object (F5 hex) ... 255
12.3.5.11 Ethernet Link Object (F6 hex) ... 257
12.3.5.12 Coupler/Controller Configuration Object (64 hex) ... 264
12.3.5.13 Discrete Input Point (65 hex) ... 266
12.3.5.14 Discrete Input Point Extended 1 (69 hex) ... 267
12.3.5.15 Discrete Input Point Extended 2 (6D hex) ... 267
12.3.5.16 Discrete Input Point Extended 3 (71 hex) ... 269
12.3.5.17 Discrete Output Point (66 hex) ... 270
12.3.5.18 Discrete Output Point Extended 1 (6A hex) ... 271
12.3.5.19 Discrete Output Point Extended 2 (6E hex) ... 272
12.3.5.20 Discrete Output Point Extended 3 (72 hex) ... 273
12.3.5.21 Analog Input Point (67 hex) ... 274
12.3.5.22 Analog Input Point Extended 1 (6B hex) ... 275
12.3.5.23 Analog Input Point Extended 2 (6F hex) ... 276
12.3.5.24 Analog Input Point Extended 3 (73 hex) ... 277
12.3.5.25 Analog Output Point (68 hex) ... 278
12.3.5.26 Analog Output Point Extended 1 (6C hex) ... 279
12.3.5.27 Analog Output Point Extended 2 (70 hex) ... 280
12.3.5.28 Analog Output Point Extended 3 (74 hex) ... 281
12.3.5.29 Module Configuration (80 hex) ... 282
12.3.5.30 Module Configuration Extended (81 hex) ... 283
12.3.5.31 Input Fieldbus Variable USINT (A0 hex) ... 284
12.3.5.32 Input Fieldbus Variable USINT Extended 1 (A1 hex) ... 285
12.3.5.33 Input Fieldbus Variable USINT Extended 2 (A2 hex) ... 286
12.3.5.34 Output Fieldbus Variable USINT (A3 hex) ... 287
12.3.5.35 Output Fieldbus Variable USINT Extended 1 (A4 hex) ... 288
12.3.5.36 Output Fieldbus Variable USINT Extended 2 (A5 hex) ... 289
12.3.5.37 Input Fieldbus Variable UINT (A6 hex) ... 290
12.3.5.38 Input Fieldbus Variable UINT Extended 1 (A7 hex) ... 291
12.3.5.39 Output Fieldbus Variable UINT (A8 hex) ... 292
12.3.5.40 Output Fieldbus Variable UINT Extended 1 (A9 hex) ... 293
12.3.5.41 Input Fieldbus Variable UDINT (AA hex) ... 294
12.3.5.42 Input Fieldbus Variable UDINT Offset (AB hex) ... 295
12.3.5.43 Output Fieldbus Variable UDINT (AC hex) ... 296

12.3.5.44 Output Fieldbus Variable UDINT Offset (AD $_{hex}$) 297

13 Busklemmen 298

13.1 Übersicht 298
13.2 Aufbau der Prozessdaten für MODBUS/TCP 299
13.2.1 Digitaleingangsklemmen 300
13.2.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose 300
13.2.1.2 2-Kanal-Digitaleingangsklemmen 300
13.2.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose 300
13.2.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten 301
13.2.1.5 4-Kanal-Digitaleingangsklemmen 301
13.2.1.6 8-Kanal-Digitaleingangsklemmen 301
13.2.1.7 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten 302
13.2.1.8 16-Kanal-Digitaleingangsklemmen 302
13.2.2 Digitalausgangsklemmen 303
13.2.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten 303
13.2.2.2 2-Kanal-Digitalausgangsklemmen 303
13.2.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 304
13.2.2.4 4-Kanal-Digitalausgangsklemmen 305
13.2.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 305
13.2.2.6 8-Kanal-Digitalausgangsklemmen 305
13.2.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 306
13.2.2.8 16-Kanal-Digitalausgangsklemmen 306
13.2.2.9 8-Kanal-Digitaleingangsklemmen/-Digitalausgangsklemmen 307
13.2.3 Analogeingangsklemmen 308
13.2.3.1 1-Kanal-Analogeingangsklemmen 308
13.2.3.2 2-Kanal-Analogeingangsklemmen 308
13.2.3.3 4-Kanal-Analogeingangsklemmen 309
13.2.3.4 3-Phasen-Leistungsmessklemme 310
13.2.3.5 8-Kanal-Analogeingangsklemmen 311
13.2.4 Analogausgangsklemmen 312
13.2.4.1 2-Kanal-Analogausgangsklemmen 312
13.2.4.2 4-Kanal-Analogausgangsklemmen 312
13.2.4.3 8-Kanal-Analogausgangsklemmen 313
13.2.5 Sonderklemmen 314
13.2.5.1 Zählerklemmen 314
13.2.5.2 Pulsweitenklemmen 316
13.2.5.3 Serielle Schnittstellen mit alternativem Datenformat 316
13.2.5.4 Serielle Schnittstellen mit Standard-Datenformat 317
13.2.5.5 Datenaustauschklemmen 317
13.2.5.6 SSI-Geber-Interface-Busklemmen 317
13.2.5.7 Weg- und Winkelmessung 318
13.2.5.8 DC-Drive Controller 320
13.2.5.9 Steppercontroller 321
13.2.5.10 RTC-Modul 322
13.2.5.11 DALI/DSI-Masterklemme 322

13.2.5.12 DALI-Multi-Master-Klemme ... 323
13.2.5.13 LON®-FTT-Klemme ... 325
13.2.5.14 Funkreceiver EnOcean ... 325
13.2.5.15 MP-Bus-Masterklemme ... 325
13.2.5.16 *Bluetooth*® RF-Transceiver ... 326
13.2.5.17 Schwingstärke/Wälzlagerüberwachung VIB I/O ... 327
13.2.5.18 KNX/EIB/TP1-Klemme ... 327
13.2.5.19 AS-Interface-Masterklemme ... 328
13.2.6 Systemklemmen ... 329
13.2.6.1 Systemklemmen mit Diagnose ... 329
13.2.6.2 Binäre Platzhalterklemmen ... 329
13.3 Aufbau der Prozessdaten für EtherNet/IP ... 330
13.3.1 Digitaleingangsklemmen ... 331
13.3.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose ... 331
13.3.1.2 2-Kanal-Digitaleingangsklemmen ... 331
13.3.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose ... 332
13.3.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten ... 332
13.3.1.5 4-Kanal-Digitaleingangsklemmen ... 332
13.3.1.6 8-Kanal-Digitaleingangsklemmen ... 333
13.3.1.7 16-Kanal-Digitaleingangsklemmen ... 333
13.3.2 Digitalausgangsklemmen ... 334
13.3.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten ... 334
13.3.2.2 2-Kanal-Digitalausgangsklemmen ... 335
13.3.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten ... 335
13.3.2.4 4-Kanal-Digitalausgangsklemmen ... 336
13.3.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten ... 337
13.3.2.6 8-Kanal-Digitalausgangsklemmen ... 337
13.3.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten ... 338
13.3.2.8 16-Kanal-Digitalausgangsklemmen ... 338
13.3.2.9 8-Kanal-Digitalein- / -ausgangsklemmen ... 339
13.3.3 Analogeingangsklemmen ... 340
13.3.3.1 1-Kanal-Analogeingangsklemmen ... 340
13.3.3.2 2-Kanal-Analogeingangsklemmen ... 341
13.3.3.3 4-Kanal-Analogeingangsklemmen ... 341
13.3.3.4 4-Kanal-Analogeingangsklemmen für Widerstandssensoren (RTD) 341
13.3.3.5 3-Phasen-Leistungsmessklemme ... 342
13.3.3.6 8-Kanal-Analogeingangsklemmen ... 343
13.3.4 Analogausgangsklemmen ... 343
13.3.4.1 2-Kanal-Analogausgangsklemmen ... 343
13.3.4.2 4-Kanal-Analogausgangsklemmen ... 344
13.3.4.3 8-Kanal-Analogausgangsklemmen ... 344
13.3.5 Sonderklemmen ... 345
13.3.5.1 Zählerklemmen ... 345
13.3.5.2 Pulsweitenklemmen ... 347
13.3.5.3 Serielle Schnittstellen mit alternativem Datenformat ... 347
13.3.5.4 Serielle Schnittstellen mit Standard Datenformat ... 348

13.3.5.5 Datenaustauschklemmen ... 349
13.3.5.6 SSI-Geber Interface Busklemmen ... 349
13.3.5.7 Weg- und Winkelmessung ... 350
13.3.5.8 DC-Drive Controller ... 352
13.3.5.9 Steppercontroller ... 353
13.3.5.10 RTC-Modul ... 354
13.3.5.11 DALI/DSI-Masterklemme ... 354
13.3.5.12 Funkreceiver EnOcean ... 355
13.3.5.13 MP-Bus-Masterklemme ... 355
13.3.5.14 *Bluetooth*® RF-Transceiver ... 356
13.3.5.15 Schwingstärke/Wälzlagerüberwachung VIB I/O ... 357
13.3.5.16 AS-interface Masterklemme ... 358
13.3.6 Systemklemmen ... 359
13.3.6.1 Systemklemmen mit Diagnose ... 359
13.3.6.2 Binäre Platzhalterklemme ... 359

14 Anwendungsbeispiele ... 360

14.1 Test von MODBUS-Protokoll und Feldbusknoten ... 360
14.2 Visualisierung und Steuerung mittels SCADA-Software ... 360

15 Einsatz in explosionsgefährdeten Bereichen ... 363

15.1 Beispielhafter Aufbau der Kennzeichnung ... 364
15.1.1 Kennzeichnung für Europa gemäß ATEX und IEC-Ex ... 364
15.1.2 Kennzeichnung für Amerika gemäß NEC 500 ... 369
15.2 Errichtungsbestimmungen ... 370
15.2.1 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 07 ATEX 554086 X) ... 371
15.2.2 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 12 ATEX 106032 X) ... 372
15.2.3 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 09.0001 X) ... 373
15.2.4 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 12.0039 X) ... 374
15.2.5 Besondere Bedingungen für den sicheren Betrieb nach ANSI/ISA 12.12.01 ... 375

Abbildungsverzeichnis ... 376

Tabellenverzeichnis ... 378

1 Hinweise zu dieser Dokumentation

Hinweis

Dokumentation aufbewahren!
Diese Dokumentation ist Teil des Produkts. Bewahren Sie deshalb die Dokumentation während der gesamten Nutzungsdauer des Produkts auf. Geben Sie die Dokumentation an jeden nachfolgenden Benutzer des Produkts weiter. Stellen Sie darüber hinaus sicher, dass gegebenenfalls jede erhaltene Ergänzung in die Dokumentation mit aufgenommen wird.

1.1 Gültigkeitsbereich

Die vorliegende Dokumentation gilt für den „ETHERNET ECO Controller“ (750-852).

Das Produkt „ETHERNET ECO Controller“ (750-852) darf nur nach Anweisungen dieser Betriebsanleitung und der Systembeschreibung zum WAGO-I/O-SYSTEM 750 installiert und betrieben werden.

ACHTUNG

Versorgungsauslegung des WAGO-I/O-SYSTEMs 750 beachten!
Sie benötigen zusätzlich zu dieser Betriebsanleitung die Systembeschreibung zum WAGO-I/O-SYSTEM 750, die unter www.wago.com herunterzuladen ist. Dort erhalten Sie unter anderem wichtige Informationen zu Potentialtrennung, Systemversorgung und Einspeisungsvorschriften.

1.2 Urheberschutz

Diese Dokumentation, einschließlich aller darin befindlichen Abbildungen, ist urheberrechtlich geschützt. Jede Weiterverwendung dieser Dokumentation, die von den urheberrechtlichen Bestimmungen abweicht, ist nicht gestattet. Die Reproduktion, Übersetzung in andere Sprachen sowie die elektronische und fototechnische Archivierung und Veränderung bedarf der schriftlichen Genehmigung der WAGO Kontakttechnik GmbH & Co. KG, Minden. Zuwiderhandlungen ziehen einen Schadenersatzanspruch nach sich.

1.3 Symbole

GEFAHR	**Warnung vor Personenschäden!** Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.
GEFAHR	**Warnung vor Personenschäden durch elektrischen Strom!** Kennzeichnet eine unmittelbare Gefährdung mit hohem Risiko, die Tod oder schwere Körperverletzung zur Folge haben wird, wenn sie nicht vermieden wird.
WARNUNG	**Warnung vor Personenschäden!** Kennzeichnet eine mögliche Gefährdung mit mittlerem Risiko, die Tod oder (schwere) Körperverletzung zur Folge haben kann, wenn sie nicht vermieden wird.
VORSICHT	**Warnung vor Personenschäden!** Kennzeichnet eine mögliche Gefährdung mit geringem Risiko, die leichte oder mittlere Körperverletzung zur Folge haben könnte, wenn sie nicht vermieden wird.
ACHTUNG	**Warnung vor Sachschäden!** Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.
ESD	**Warnung vor Sachschäden durch elektrostatische Aufladung!** Kennzeichnet eine mögliche Gefährdung, die Sachschaden zur Folge haben könnte, wenn sie nicht vermieden wird.
Hinweis	**Wichtiger Hinweis!** Kennzeichnet eine mögliche Fehlfunktion, die aber keinen Sachschaden zur Folge hat, wenn sie nicht vermieden wird.
Information	**Weitere Information** Weist auf weitere Informationen hin, die kein wesentlicher Bestandteil dieser Dokumentation sind (z. B. Internet).

1.4 Darstellung der Zahlensysteme

Tabelle 1: Darstellungen der Zahlensysteme

Zahlensystem	Beispiel	Bemerkung
Dezimal	100	Normale Schreibweise
Hexadezimal	0x64	C-Notation
Binär	'100' '0110.0100'	In Hochkomma, Nibble durch Punkt getrennt

1.5 Schriftkonventionen

Tabelle 2: Schriftkonventionen

Schriftart	Bedeutung
kursiv	Namen von Pfaden und Dateien werden kursiv dargestellt z. B.: *C:\Programme\WAGO Software*
Menü	Menüpunkte werden fett dargestellt z. B.: **Speichern**
>	Ein „Größer als“- Zeichen zwischen zwei Namen bedeutet die Auswahl eines Menüpunktes aus einem Menü z. B.: **Datei > Neu**
Eingabe	Bezeichnungen von Eingabe- oder Auswahlfeldern werden fett dargestellt z. B.: **Messbereichsanfang**
„Wert“	Eingabe- oder Auswahlwerte werden in Anführungszeichen dargestellt z. B.: Geben Sie unter **Messbereichsanfang** den Wert „4 mA“ ein.
[Button]	Schaltflächenbeschriftungen in Dialogen werden fett dargestellt und in eckige Klammern eingefasst z. B.: **[Eingabe]**
[Taste]	Tastenbeschriftungen auf der Tastatur werden fett dargestellt und in eckige Klammern eingefasst z. B.: **[F5]**

2 Wichtige Erläuterungen

Dieses Kapitel beinhaltet ausschließlich eine Zusammenfassung der wichtigsten Sicherheitsbestimmungen und Hinweise. Diese werden in den einzelnen Kapiteln wieder aufgenommen. Zum Schutz vor Personenschäden und zur Vorbeugung von Sachschäden an Geräten ist es notwendig, die Sicherheitsrichtlinien sorgfältig zu lesen und einzuhalten.

2.1 Rechtliche Grundlagen

2.1.1 Änderungsvorbehalt

Die WAGO Kontakttechnik GmbH & Co. KG behält sich Änderungen, die dem technischen Fortschritt dienen, vor. Alle Rechte für den Fall der Patenterteilung oder des Gebrauchsmusterschutzes sind der WAGO Kontakttechnik GmbH & Co. KG vorbehalten. Fremdprodukte werden stets ohne Vermerk auf Patentrechte genannt. Die Existenz solcher Rechte ist daher nicht auszuschließen.

2.1.2 Personalqualifikation

Sämtliche Arbeitsschritte, die an den Geräten des WAGO-I/O-SYSTEMs 750 durchgeführt werden, dürfen nur von Elektrofachkräften mit ausreichenden Kenntnissen im Bereich der Automatisierungstechnik vorgenommen werden. Diese müssen mit den aktuellen Normen und Richtlinien für die Geräte und das Automatisierungsumfeld vertraut sein.

Alle Eingriffe in die Steuerung sind stets von Fachkräften mit ausreichenden Kenntnissen in der SPS-Programmierung durchzuführen.

2.1.3 Bestimmungsgemäße Verwendung des WAGO-I/O-SYSTEMs 750

Feldbuskoppler, Feldbuscontroller und Busklemmen des modularen WAGO-I/O-SYSTEMs 750 dienen dazu, digitale und analoge Signale von Sensoren aufzunehmen und an Aktoren auszugeben oder an übergeordnete Steuerungen weiterzuleiten. Mit den programmierbaren Feldbuscontrollern ist zudem eine (Vor-)Verarbeitung möglich.

Die Geräte sind für ein Arbeitsumfeld entwickelt, welches der Schutzklasse IP20 genügt. Es besteht Fingerschutz und Schutz gegen feste Fremdkörper bis 12,5 mm, jedoch kein Schutz gegen Wasser. Der Betrieb der Geräte in nasser und staubiger Umgebung ist nicht gestattet, sofern nicht anders angegeben.

Der Betrieb von Geräten des WAGO-I/O-SYSTEMs 750 im Wohnbereich ist ohne weitere Maßnahmen nur zulässig, wenn diese die Emissionsgrenzen (Störaussendungen) gemäß EN 61000-6-3 einhalten. Entsprechende Angaben finden Sie im Kapitel „Gerätebeschreibung“ > „Normen und Richtlinien“ im Handbuch zum eingesetzten Feldbuskoppler/-controller.

Für den Betrieb des WAGO-I/O-SYSTEMs 750 in explosionsgefährdeten Bereichen ist ein entsprechender Gehäuseschutz gemäß der Richtlinie 94/9/EG erforderlich. Zusätzlich ist zu beachten, dass eine Baumusterprüfbescheinigung erwirkt werden muss, die den korrekten Einbau des Systems im Gehäuse bzw. Schaltschrank bestätigt.

2.1.4 Technischer Zustand der Geräte

Die Geräte werden ab Werk für den jeweiligen Anwendungsfall mit einer festen Hard- und Software-Konfiguration ausgeliefert. Alle Veränderungen an der Hard- oder Software sowie der nicht bestimmungsgemäße Gebrauch der Komponenten bewirken den Haftungsausschluss der WAGO Kontakttechnik GmbH & Co. KG.

Wünsche an eine abgewandelte bzw. neue Hard- oder Software-Konfiguration richten Sie bitte an die WAGO Kontakttechnik GmbH & Co. KG.

2.2 Sicherheitshinweise

Beim Einbauen des Gerätes in Ihre Anlage und während des Betriebes sind folgende Sicherheitshinweise zu beachten:

GEFAHR

Nicht an Geräten unter Spannung arbeiten!
Schalten Sie immer alle verwendeten Spannungsversorgungen für das Gerät ab, bevor Sie es montieren, Störungen beheben oder Wartungsarbeiten vornehmen.

GEFAHR

Nur in Gehäusen, Schränken oder elektrischen Betriebsräumen einbauen!
Das WAGO-I/O-SYSTEM 750 mit seinen Geräten ist ein offenes Betriebsmittel. Bauen Sie dieses ausschließlich in abschließbaren Gehäusen, Schränken oder in elektrischen Betriebsräumen auf. Ermöglichen Sie nur autorisiertem Fachpersonal den Zugang mittels Schlüssel oder Werkzeug.

GEFAHR

Unfallverhütungsvorschriften beachten!
Beachten Sie bei der Montage, Inbetriebnahme, Wartung und Störbehebung die für Ihre Maschine zutreffenden Unfallverhütungsvorschriften wie beispielsweise die BGV A3 „Elektrische Anlagen und Betriebsmittel“.

GEFAHR

Auf normgerechten Anschluss achten!
Zur Vermeidung von Gefahren für das Personal und Störungen an Ihrer Anlage, verlegen Sie die Daten- und Versorgungsleitungen normgerecht und achten Sie auf die korrekte Anschlussbelegung. Beachten Sie die für Ihre Anwendung zutreffenden EMV-Richtlinien.

ACHTUNG

Defekte oder beschädigte Geräte austauschen!
Tauschen Sie defekte oder beschädigte Geräte (z. B. bei deformierten Kontakten) aus, da die Funktion der betroffenen Geräte langfristig nicht sichergestellt ist.

ACHTUNG

Geräte vor kriechenden und isolierenden Stoffen schützen!
Die Geräte sind unbeständig gegen Stoffe, die kriechende und isolierende Eigenschaften besitzen, z. B. Aerosole, Silikone, Triglyceride (Bestandteil einiger Handcremes). Sollten Sie nicht ausschließen können, dass diese Stoffe im Umfeld der Geräte auftreten, bauen Sie die Geräte in ein Gehäuse ein, das resistent gegen oben genannte Stoffe ist. Verwenden Sie generell zur Handhabung der Geräte saubere Werkzeuge und Materialien.

ACHTUNG

Nur mit zulässigen Materialien reinigen!
Reinigen Sie verschmutzte Kontakte mit ölfreier Druckluft oder mit Spiritus und einem Ledertuch.

ACHTUNG

Kein Kontaktspray verwenden!
Verwenden Sie kein Kontaktspray, da in Verbindung mit Verunreinigungen die Funktion der Kontaktstelle beeinträchtigt werden kann.

ACHTUNG

Verpolungen vermeiden!
Vermeiden Sie die Verpolung der Daten- und Versorgungsleitungen, da dies zu Schäden an den Geräten führen kann.

ESD

Elektrostatische Entladung vermeiden!
In den Geräten sind elektronische Komponenten integriert, die Sie durch elektrostatische Entladung bei Berührung zerstören können. Beachten Sie die Sicherheitsmaßnahmen gegen elektrostatische Entladung gemäß DIN EN 61340-5-1/-3. Achten Sie beim Umgang mit den Geräten auf gute Erdung der Umgebung (Personen, Arbeitsplatz und Verpackung).

2.3 Spezielle Einsatzbestimmungen für ETHERNET-Geräte

Wo nicht speziell beschrieben, sind ETHERNET-Geräte für den Einsatz in lokalen Netzwerken bestimmt. Beachten Sie folgende Hinweise, wenn Sie ETHERNET-Geräte in Ihrer Anlage einsetzen:

- Verbinden Sie Steuerungskomponenten und Steuerungsnetzwerke nicht mit einem offenen Netzwerk wie dem Internet oder einem Büronetzwerk. WAGO empfiehlt, Steuerungskomponenten und Steuerungsnetzwerke hinter einer Firewall anzubringen.
- Beschränken Sie den physikalischen und elektronischen Zugang zu sämtlichen Automatisierungskomponenten auf einen autorisierten Personenkreis.
- Ändern Sie vor der ersten Inbetriebnahme unbedingt die standardmäßig eingestellten Passwörter! Sie verringern so das Risiko, dass Unbefugte Zugriff auf Ihr System erhalten.
- Ändern Sie regelmäßig die verwendeten Passwörter! Sie verringern so das Risiko, dass Unbefugte Zugriff auf Ihr System erhalten.
- Ist ein Fernzugriff auf Steuerungskomponenten und Steuerungsnetzwerke erforderlich, sollte ein „Virtual Private Network“ (VPN) genutzt werden.
- Führen Sie regelmäßig eine Bedrohungsanalyse durch. So können Sie prüfen, ob die getroffenen Maßnahmen Ihrem Schutzbedürfnis entsprechen.
- Wenden Sie in der sicherheitsgerichteten Gestaltung Ihrer Anlage „Defense-in-depth“-Mechanismen an, um den Zugriff und die Kontrolle auf individuelle Produkte und Netzwerke einzuschränken.

3 Systembeschreibung

Das WAGO-I/O-SYSTEM 750 ist ein modulares und feldbusunabhängiges Ein-/ Ausgabesystem (E/A-System). Der hier beschriebene Aufbau besteht aus einem Feldbuskoppler/-controller (1) und den angereihten Busklemmen (2) für beliebige Signalformen, die zusammen den Feldbusknoten bilden. Die Endklemme (3) schließt den Knoten ab und ist für den ordnungsgemäßen Betrieb des Feldbusknotens zwingend erforderlich.

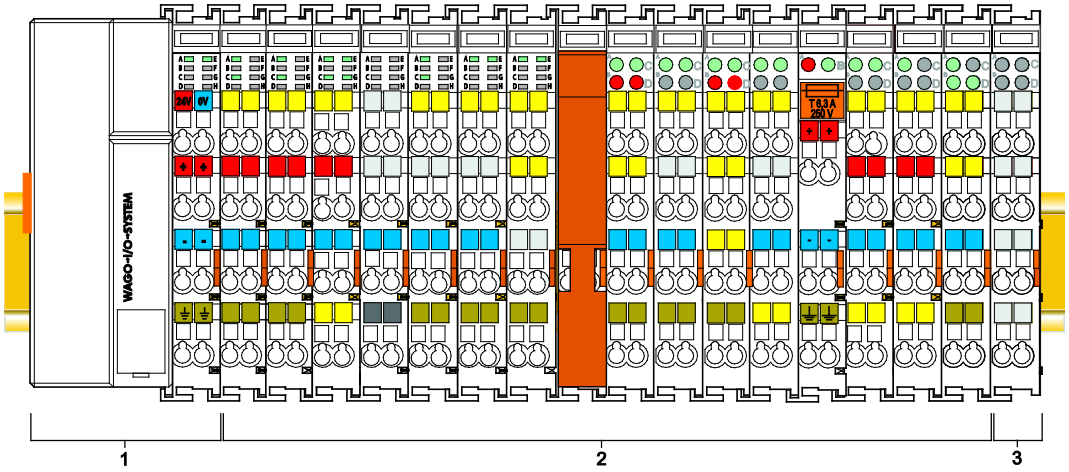


Abbildung 1: Feldbusknoten (Beispiel)

Feldbuskoppler/-controller stehen für diverse Feldbussysteme zur Verfügung.

Der ECO-Feldbuskoppler enthält ein Feldbus-Interface, eine Elektronik und eine Spannungsversorgung für das System. Das Feldbus-Interface bildet die physikalische Schnittstelle zum jeweiligen Feldbussystem. Die Elektronik verarbeitet die Daten der Busklemmen und stellt diese für die Feldbuskommunikation bereit.

An den Feldbuskoppler/-controller können Busklemmen für unterschiedliche digitale und analoge E/A-Signale sowie Sonderfunktionen angereiht werden. Die Kommunikation zwischen Feldbuskoppler/-controller und Busklemmen erfolgt über einen internen Klemmenbus (K-Bus).

Die Komponenten des WAGO-I/O-SYSTEMs 750 besitzen eine übersichtliche Anschlussebene, Leuchtdioden für die Statusanzeige, einsteckbare Mini-WSB-Schilder und Gruppenbeschriftungsschilder für die Beschriftung.

Die 1-, 2- oder 3-Leitertechnik erlaubt eine direkte Sensor- bzw. Aktorverdrahtung.

3.1 Fertigungsnummer

Die Fertigungsnummer gibt den Auslieferungszustand direkt nach der Herstellung an. Diese Nummer ist Teil der seitlichen Bedruckung jeder Komponente. Zusätzlich wird die Fertigungsnummer auf die Abdeckklappe der Konfigurations- und Programmierschnittstelle des Feldbuskopplers/-controllers gedruckt, damit sie auch im eingebauten Zustand abgelesen werden kann.

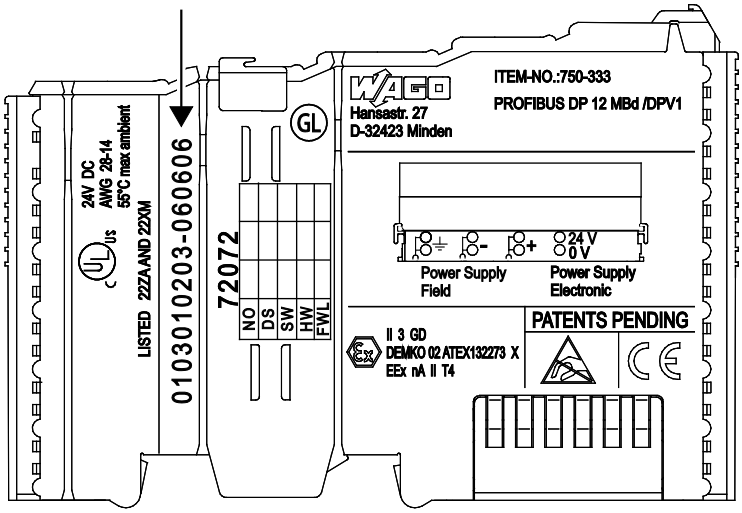


Abbildung 2: Beispiel einer seitlichen Gehäusebedruckung

Fertigungsnummer

01	**03**	**01**	**02**	**03**	**-**	**060606**
Kalender-woche	Jahr	Software-Version	Hardware-Version	Firmware-Loader-Version		Interne Nummer

Abbildung 3: Beispiel einer Fertigungsnummer

Die Fertigungsnummer setzt sich zusammen aus Herstellungswoche und -jahr, Software-Version (optional), Hardware-Version, Firmware-Loader-Version (optional) und weiteren internen Informationen der WAGO Kontakttechnik GmbH & Co. KG.

3.2 Hardware-Adresse (MAC-ID)

Das Gerät ETHERNET ECO Controller trägt eine weltweit eindeutige physikalische Adresse, die MAC-ID (Media-Access-Control-Identity).

Als Teil der seitlichen Bedruckung auf der rechten Geräteseite ist diese innerhalb des Prinzipschaltbildes des Feldbuskopplers/-controllers aufgedruckt. Zusätzlich befindet sich die MAC-ID auf einem Papierstreifen mit zwei selbstklebenden Abreißetiketten auf der linken Seite des Feldbuskopplers/-controllers.
Die MAC-ID besitzt eine feste Länge von 6 Byte (48 Bit) in hexadezimaler Schreibweise. Die ersten 3 Byte geben Auskunft über den Hersteller (z. B. 00:30:DE für WAGO). Die weiteren 3 Byte beinhalten die laufende Seriennummer der Geräte-Hardware.

3.3 Komponenten-Update

Für den Fall des Updates einer Komponente enthält die seitliche Bedruckung jeder Komponente eine vorbereitete Matrix.

Diese Matrix stellt für insgesamt drei Updates Spalten zum Eintrag der aktuellen Update-Daten zur Verfügung, wie Betriebsauftrags (BA) -Nummer (NO; ab KW 13/2004), Update-Datum (DS), Software-Version (SW, optional), Hardware-Version (HW) und die Firmware-Loader-Version (FWL, optional).

Aktuelle Versionsangabe für		1. Update	2. Update	3. Update	
BA-Nummer	**NO**				← ab KW 13/2004
Update-Datum	**DS**				
Software-Version	**SW**				
Hardware-Version	**HW**				
Firmware-Loader-Version	**FWL**				← nur Feldbuskoppler/ -controller

Ist das Update einer Komponente erfolgt, werden die aktuellen Versionsangaben in die Spalten der Matrix eingetragen.

Zusätzlich wird bei dem Update eines Feldbuskopplers/-controllers auch die Abdeckklappe der Konfigurationsschnittstelle mit der aktuellen Fertigungs- und Betriebsauftragsnummer bedruckt.

Die ursprünglichen Fertigungsangaben auf dem Gehäuse der Komponente bleiben dabei erhalten.

3.4 Lagerung, Kommissionierung und Transport

Die Komponenten sind möglichst in der Originalverpackung zu lagern. Ebenso bietet die Originalverpackung beim Transport den optimalen Schutz.

Beim Kommissionieren, Ein- und Auspacken dürfen die Kontakte nicht verschmutzt oder beschädigt werden. Die Komponenten müssen unter Beachtung der ESD-Hinweise in geeigneten Behältern/Verpackungen gelagert und transportiert werden.

3.5 Aufbaurichtlinien und Normen

DIN 60204	Elektrische Ausrüstung von Maschinen
DIN EN 50178	Ausrüstung von Starkstromanlagen mit elektronischen Betriebsmitteln (Ersatz für VDE 0160)
EN 60439	Niederspannung – Schaltgerätekombinationen

3.6 Spannungsversorgung

3.6.1 Potentialtrennung

Innerhalb des Feldbusknotens bestehen drei galvanisch getrennte Potentialgruppen:

- galvanisch getrenntes Feldbus-Interface
- Elektronik des Feldbuskopplers/-controllers und der Busklemmen (Klemmenbus)
- Alle Busklemmen besitzen eine galvanische Trennung zwischen der Systemelektronik (Klemmenbus, Logik) und der feldseitigen Elektronik. Bei einigen digitalen und analogen Eingangsklemmen ist diese Trennung kanalweise aufgebaut, siehe Katalog.

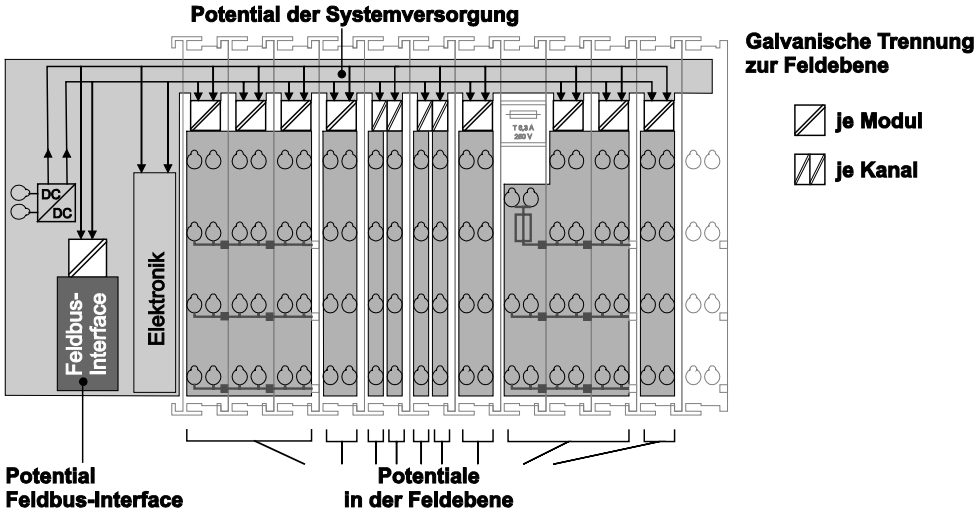


Abbildung 4: Potentialtrennung (Beispiel)

3.6.2 Systemversorgung

3.6.2.1 Anschluss

Das WAGO-I/O-SYSTEM 750 benötigt als Systemversorgung eine 24V-Gleichspannung. Die Einspeisung erfolgt über den Feldbuskoppler/-controller und bei Bedarf zusätzlich über Potentialeinspeiseklemmen mit Busnetzteil, Bestellnr. 750-613. Die Einspeisung ist gegen Verpolung geschützt.

Hinweis

Keine unzulässige Spannung/Frequenz aufschalten!
Schalten Sie keine unzulässigen Spannungs- oder Frequenzwerte auf. Dieses kann zur Zerstörung der Baugruppe führen.

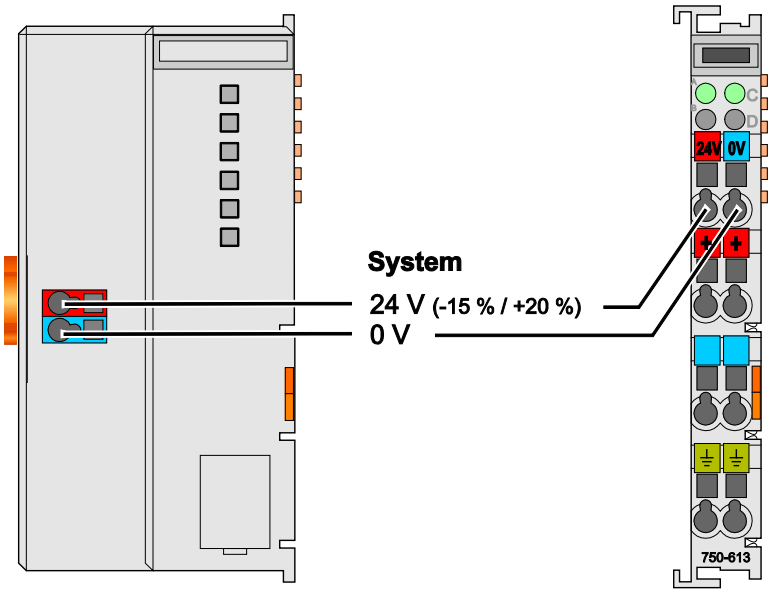


Abbildung 5: Systemversorgung

Die eingespeiste 24V-Gleichspannung versorgt alle systeminternen Bausteine, z. B. die Elektronik des Feldbuskopplers/-controllers, das Feldbus-Interface und die Busklemmen über den Klemmenbus (5V-Systemspannung). Die 5V-Systemspannung ist mit der 24V-Systemversorgung galvanisch verbunden.

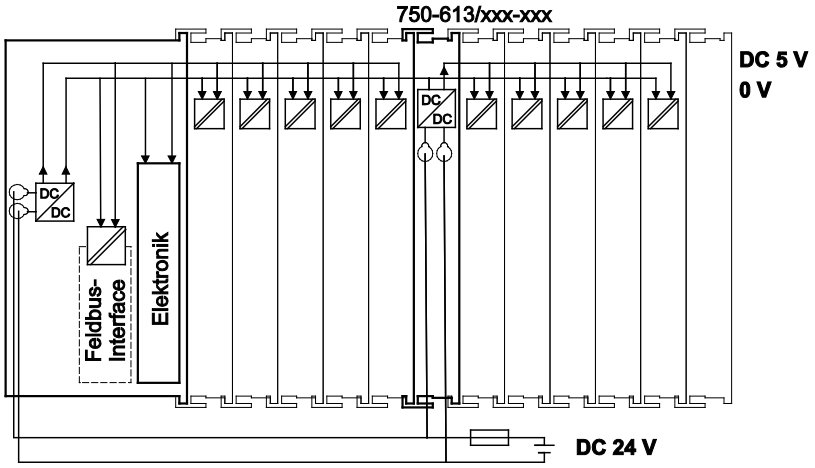


Abbildung 6: Systemspannung (Beispiel)

Hinweis

Rücksetzen des Systems nur gleichzeitig bei allen Versorgungsmodulen!
Führen Sie das Rücksetzen des Systems durch gleichzeitiges Aus- und Wiedereinschalten der Systemversorgung gleichzeitig an allen Versorgungsmodulen (Feldbuskoppler/-controller und Potentialeinspeiseklemme mit Busnetzteil) durch.

3.6.2.2 Auslegung

Hinweis

Empfehlung
Eine stabile Netzversorgung kann nicht immer und überall vorausgesetzt werden. Sie sollten daher geregelte Netzteile verwenden, um die Qualität der Versorgungsspannung zu gewährleisten.

Die Versorgungskapazität der Feldbuskoppler/-controller bzw. der Potentialeinspeiseklemme mit Busnetzteil kann den technischen Daten der Komponenten entnommen werden.

Tabelle 3: Auslegung

Interne Stromaufnahme *)	Stromaufnahme über Systemspannung (5 V für Elektronik der Busklemmen und Feldbuskoppler/-controller).
Summenstrom für Busklemmen *)	Verfügbarer Strom für die Busklemmen. Wird vom Busnetzteil bereitgestellt. Siehe Feldbuskoppler/-controller und Potentialeinspeiseklemme mit Busnetzteil

*) vgl. aktuellen Katalog, Handbücher, Internet

Beispiel:

Berechnung Stromaufnahme an einem Beispielkoppler

Interne Stromaufnahme	300 mA bei 5 V
Summenstrom für Busklemmen	700 mA bei 5 V
Summe $I_{(5\ V)\ ges}$	**1000 mA bei 5V**

Für jede Busklemme ist die interne Stromaufnahme in den technischen Daten der Busklemme angegeben. Um den Gesamtbedarf zu ermitteln, werden die Werte aller Busklemmen im Knoten summiert.

Hinweis

Summenstrom für Busklemmen beachten, evtl. Potential neu einspeisen!
Sobald die Summe der internen Stromaufnahmen der Busklemmen den Summenstrom für Busklemmen übersteigt, müssen Sie eine Potentialeinspeiseklemme mit Busnetzteil setzen. Platzieren Sie diese vor die Position, an der der zulässige Summenstrom überschritten würde.

Beispiel:

Berechnung Summenstrom an einem Beispielkoppler

In diesem Beispiel soll ein Beispielkoppler in einem Knoten mit folgenden Busklemmen eingesetzt werden:
10 Relaisklemmen (750-517) und 20 Digitaleingangsklemmen (750-405)

Interne Stromaufnahme	10 * 90 mA =	900 mA
	20 * 2 mA =	40 mA
Summe		**940 mA**

Der Beispielkoppler kann in diesem Beispiel 700 mA für die Busklemmen bereitstellen (siehe „Summenstrom für Busklemmen“ im Datenblatt). Folglich muss eine Potentialeinspeiseklemme mit Busnetzteil (750-613), z. B. in der Mitte des Knotens, vorgesehen werden.

Hinweis

Empfehlung
Sie können mit der WAGO-ProServe®-Software **smartDESIGNER** den Aufbau eines Feldbusknotens konfigurieren. Über die integrierte Plausibilitätsprüfung können Sie die Konfiguration überprüfen.

Der maximale Eingangsstrom der 24V-Systemversorgung beträgt je Einspeisestelle 500 mA.
Die genaue Stromaufnahme ($I_{(V)}$) kann mit folgenden Formeln ermittelt werden:

Feldbuskoppler oder -controller

$I_{(5\ V)\ ges.}$ = Summe aller Stromaufnahmen der angereihten Busklemmen + interne Stromaufnahme des Feldbuskopplers/-controllers

Potentialeinspeiseklemme

$I_{(5\ V)\ ges.}$ = Summe aller Stromaufnahmen der angereihten Busklemmen an der Potentialeinspeiseklemme

$$\text{Eingangsstrom } I_{(24\ V)} = \frac{5\ V}{24\ V} * \frac{I_{(5\ V)\ ges.}}{\eta}$$

η = Netzteilwirkungsgrad bei Nennlast 24 V

Hinweis

Bei Test der Stromaufnahme alle Ausgänge aktivieren!
Übersteigt die Stromaufnahme einer Einspeisestelle für die 24V-Systemversorgung 500 mA, kann die Ursache ein falsch ausgelegter Knoten oder ein Defekt sein.
Sie müssen bei dem Test alle Ausgänge aktivieren.

3.6.3 Feldversorgung

3.6.3.1 Anschluss

Sensoren und Aktoren können direkt in 1- bis 4-Leiteranschlusstechnik an den jeweiligen Kanal der Busklemmen angeschlossen werden. Die Versorgung der Sensoren und Aktoren übernimmt die Busklemme. Die Ein- und Ausgangstreiber einiger Busklemmen benötigen die feldseitige Versorgungsspannung.

Die feldseitige Versorgungsspannung wird mit Potentialeinspeiseklemme eingespeist. In diesem Fall handelt es sich um eine passive Einspeisung ohne Schutzeinrichtung. Es stehen Einspeiseklemmen für unterschiedliche Potentiale, z. B. DC 24 V oder AC 230 V, zur Verfügung.

Zur Einspeisung anderer Feldpotentiale (DC 24 V, AC/DC 0 … 230 V, AC 120 V, AC 230 V) stehen Potentialeinspeiseklemmen mit oder ohne Sicherungshalter und Diagnosemöglichkeit zur Verfügung. Mit Hilfe der Potentialeinspeiseklemmen können außerdem unterschiedliche Potentialgruppen aufgebaut werden. Die Anschlüsse sind paarweise mit einem Leistungskontakt verbunden.

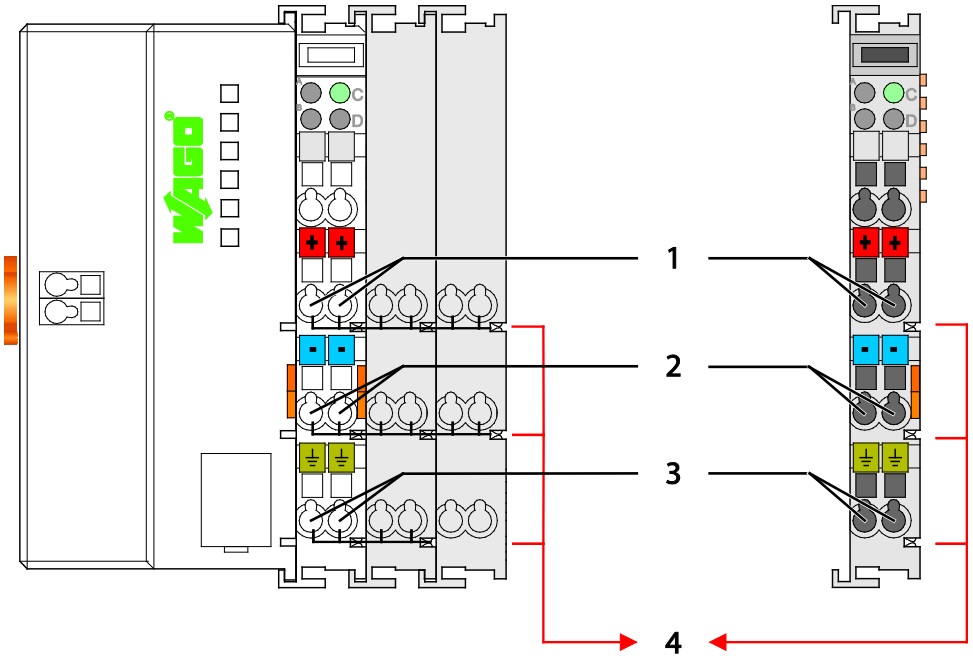


Abbildung 7: Feldversorgung (Sensor/Aktor) für ECO-Feldbuskoppler

Tabelle 4: Legende zur Abbildung „Feldversorgung (Sensor/Aktor) für ECO-Feldbuskoppler”

Feldversorgung	
1	24 V (-15 % / +20 %)
2	0 V
3	Optionales Erdpotential
Leistungskontakte	
4	Potentialverteilung zu benachbarten Busklemmen

Hinweis

Anlegen der Feldversorgung in Ausnahmefällen direkt an Busklemmen
In Ausnahmefällen kann die 24-V-Feldversorgung auch direkt an eine Busklemme angeschlossen werden, wenn die entsprechenden Klemmstellen nicht für die Peripheriegeräteversorgung benötigt werden. Die Klemmstellen benötigen in diesem Fall die Verbindung zu den Leistungskontakten.

Die Weiterleitung der Versorgungsspannung für die Feldseite erfolgt über die Leistungskontakte. Das geschieht automatisch durch Anrasten der jeweiligen Busklemme.

Die Strombelastung der Leistungskontakte darf 10 A nicht dauerhaft überschreiten.

Durch Setzen einer zusätzlichen Einspeiseklemme wird die über die Leistungskontakte geführte Feldversorgung unterbrochen. Ab dort erfolgt eine neue Einspeisung, die auch einen Potentialwechsel beinhalten kann.

Hinweis

Potential bei Unterbrechung der Leistungskontakte neu einspeisen!
Einige Busklemmen besitzen keine oder nur einzelne Leistungskontakte (abhängig von der E/A-Funktion). Dadurch wird die Weitergabe des entsprechenden Potentials unterbrochen. Wenn bei nachfolgenden Busklemmen eine Feldversorgung über die Leistungskontakte erforderlich ist, müssen Sie eine Potentialeinspeiseklemme einsetzen.
Beachten Sie die Datenblätter der einzelnen Busklemmen.

Hinweis

Bei unterschiedlichen Potentialgruppen Distanzklemme verwenden!
Bei einem Knotenaufbau mit verschiedenen Potentialgruppen, z. B. der Wechsel von DC 24 V auf AC 230 V, sollten Sie eine Distanzklemme einsetzen. Die optische Trennung der Potentiale mahnt zur Vorsicht bei Verdrahtungs- und Wartungsarbeiten. Somit können Sie die Folgen von Verdrahtungsfehlern vermeiden.

3.6.3.2 Absicherung

Die interne Absicherung der Feldversorgung ist für verschiedene Feldspannungen über entsprechende Potentialeinspeiseklemmen möglich.

Tabelle 5: Potentialeinspeiseklemmen

Bestellnummer	Feldspannung
750-601	24 V DC, Einspeisung/Sicherung
750-609	230 V AC, Einspeisung/Sicherung
750-615	120 V AC, Einspeisung/Sicherung
750-617	24 V AC, Einspeisung/Sicherung
750-610	24 V DC, Einspeisung/Sicherung/Diagnose
750-611	230 V AC, Einspeisung/Sicherung/Diagnose
750-606	Potentialeinspeisung DC 24 V, 1,0 A, Ex i
750-625/000-001	Potentialeinspeisung DC 24 V, 1,0 A, Ex i (ohne Diagnose)

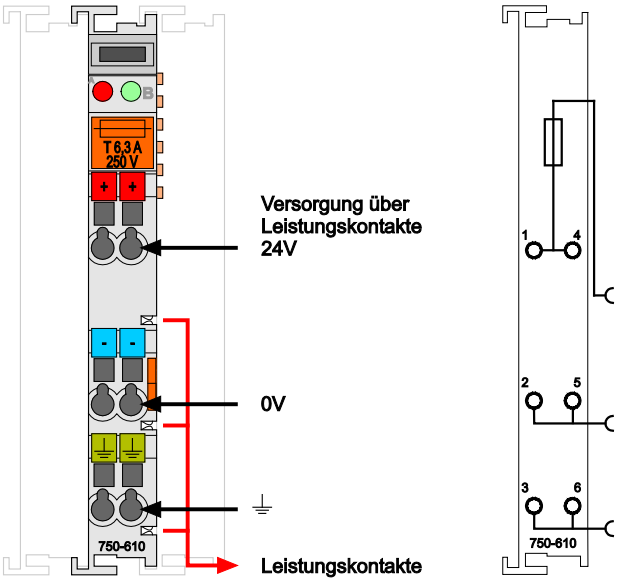


Abbildung 8: Potentialeinspeiseklemme mit Sicherungshalter (Beispiel 750-610)

ACHTUNG

Auf max. Verlustleistung und ggf. UL-Zulassung achten!
Bei Einspeiseklemmen mit Sicherungshalter dürfen Sie nur Sicherungen mit einer max. Verlustleitung von 1,6 W (IEC 127) einsetzen. Bei Anlagen, die eine UL-Zulassung besitzen, achten Sie zusätzlich darauf, dass Sie nur UL-zugelassene Sicherungen verwenden.

Um eine Sicherung einzulegen, zu wechseln oder um nachfolgende Busklemmen spannungsfrei zu schalten, kann der Sicherungshalter herausgezogen werden. Dazu wird, z. B. mit einem Schraubendreher, in einen der beidseitig vorhandenen Schlitze gegriffen und der Halter herausgezogen.

Abbildung 9: Sicherungshalter ziehen

Der Sicherungshalter wird geöffnet, indem die Abdeckung zur Seite geklappt wird.

Abbildung 10: Sicherungshalter öffnen

Abbildung 11: Sicherung wechseln

Nach dem Sicherungswechsel wird der Sicherungshalter in seine ursprüngliche Position zurückgeschoben.

Alternativ kann die Absicherung extern erfolgen. Hierbei bieten sich die Sicherungsklemmen der WAGO-Serien 281 und 282 an.

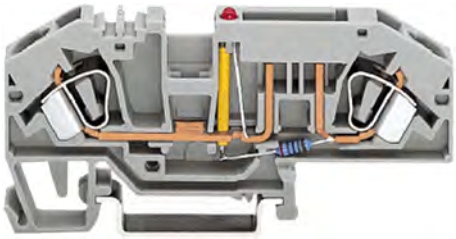


Abbildung 12: Sicherungsklemmen für Kfz-Sicherungen, Serie 282

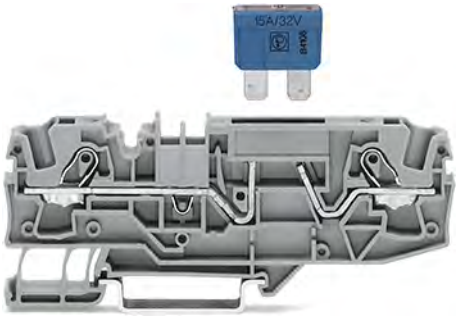


Abbildung 13: Sicherungsklemmen für Kfz-Sicherungen, Serie 2006

Abbildung 14: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 281

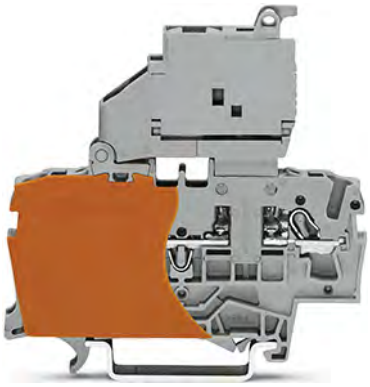


Abbildung 15: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 2002

3.6.4 Versorgungsbeispiel

Hinweis

System- und Feldversorgung getrennt einspeisen!
Speisen Sie die Systemversorgung und die Feldversorgung getrennt ein, um bei aktorseitigen Kurzschlüssen den Busbetrieb zu gewährleisten.

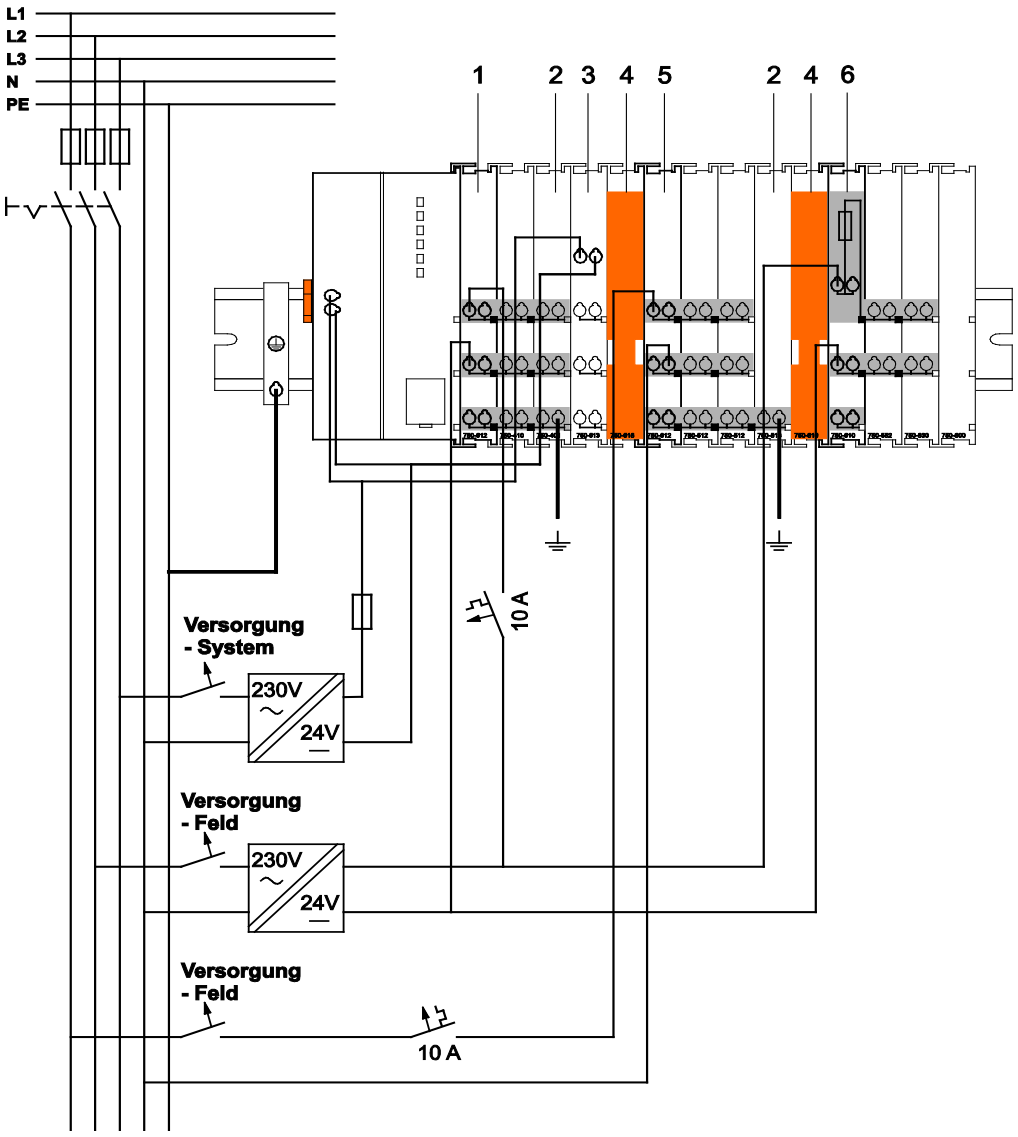


Abbildung 16: Versorgungsbeispiel

Tabelle 6: Legende zur Abbildung „Versorgungsbeispiel für Feldbuskoppler/-controller”

Pos.	Beschreibung
1	Potentialeinspeisung am Feldbuskoppler/-controller über externe Einspeiseklemme
2	Potentialeinspeisung mit jeweils optionaler Funktionserde
3	Potentialeinspeisung mit Busnetzteil
4	Distanzklemme empfohlen
5	Potentialeinspeisung passiv
6	Potentialeinspeisung mit Sicherungshalter/Diagnose

3.6.5 Netzgeräte

Das WAGO-I/O-SYSTEM 750 benötigt zum Betrieb eine 24V-Gleichspannung (Systemversorgung).

Hinweis **Empfehlung**

Eine stabile Netzversorgung kann nicht immer und überall vorausgesetzt werden. Daher sollten Sie geregelte Netzteile verwenden, um die Qualität der Versorgungsspannung zu gewährleisten (siehe auch Tabelle „WAGO-Netzgeräte“).

Für kurze Spannungseinbrüche ist ein Puffer (200 µF pro 1 A Laststrom) einzuplanen.

Hinweis **Netzausfallzeit nicht nach IEC 61131-2!**

Beachten Sie, dass die Netzausfallzeit von 10 ms nach IEC 61131-2 in einem Maximalausbau nicht eingehalten wird.

Je Einspeisestelle für die Feldversorgung ist der Strombedarf individuell zu ermitteln. Dabei sind alle Lasten durch Feldgeräte und Busklemmen zu berücksichtigen. Die Feldversorgung hat ebenfalls Einfluss auf die Busklemmen, da die Ein- und Ausgangstreiber einiger Busklemmen die Spannung der Feldversorgung benötigen.

Hinweis **System- und Feldversorgung getrennt einspeisen!**

Speisen Sie die Systemversorgung und die Feldversorgung getrennt ein, um bei aktorseitigen Kurzschlüssen den Busbetrieb zu gewährleisten.

Tabelle 7: WAGO-Netzgeräte (Auswahl)

WAGO-Netzgeräte Bestellnummer	Beschreibung
787-612	Primär getaktet; DC 24 V; 2,5 A Eingangsspannung AC 230 V
787-622	Primär getaktet, DC 24 V; 5 A Eingangsspannung AC 230 V
787-632	Primär getaktet; DC 24 V; 10 A Eingangsspannungsbereich AC 230/115 V
	Schienenmontierbare Netzgeräte auf Universal Montagesockel
288-809	AC 115 V/DC 24 V; 0,5 A
288-810	AC 230 V/DC 24 V; 0,5 A
288-812	AC 230 V/DC 24 V; 2 A
288-813	AC 115 V/DC 24 V; 2 A

3.7 Erdung

3.7.1 Erdung der Tragschiene

3.7.1.1 Rahmenaufbau

Beim Rahmenaufbau ist die Tragschiene mit dem elektrisch leitenden Schrankrahmen bzw. Gehäuse verschraubt. Der Rahmen bzw. das Gehäuse muss geerdet sein. Über die Verschraubung wird auch die elektrische Verbindung hergestellt. Somit ist die Tragschiene geerdet.

GEFAHR

Auf ausreichende Erdung achten!
Achten Sie auf eine einwandfreie elektrische Verbindung zwischen der Tragschiene und dem Rahmen bzw. Gehäuse, um eine ausreichende Erdung sicher zu stellen.

3.7.1.2 Isolierter Aufbau

Ein isolierter Aufbau liegt dann vor, wenn es konstruktiv keine direkte leitende Verbindung zwischen Schrankrahmen oder Maschinenteilen und der Tragschiene gibt. Hier muss über einen elektrischen Leiter entsprechend geltender nationaler Sicherheitsvorschriften die Erdung aufgebaut werden.

Hinweis

Empfehlung
Der optimale Aufbau ist eine metallische Montageplatte mit Erdungsanschluss, die elektrisch leitend mit der Tragschiene verbunden ist.

Die separate Erdung der Tragschiene kann einfach mit Hilfe der WAGO-Schutzleiterklemmen aufgebaut werden.

Tabelle 8: WAGO-Schutzleiterklemmen

Bestellnummer	Beschreibung
283-609	1-Leiter-Schutzleiterklemme kontaktiert den Schutzleiter direkt auf der Tragschiene; Anschlussquerschnitt: 0,2 mm² … 16 mm² **Hinweis:** Abschlussplatte (283-320) mitbestellen

3.7.2 Funktionserde

Die Funktionserde erhöht die Störfestigkeit gegenüber elektromagnetischen Einflüssen. Einige Komponenten des I/O-Systems besitzen einen Tragschienenkontakt, der elektromagnetische Störungen zur Tragschiene ableitet.

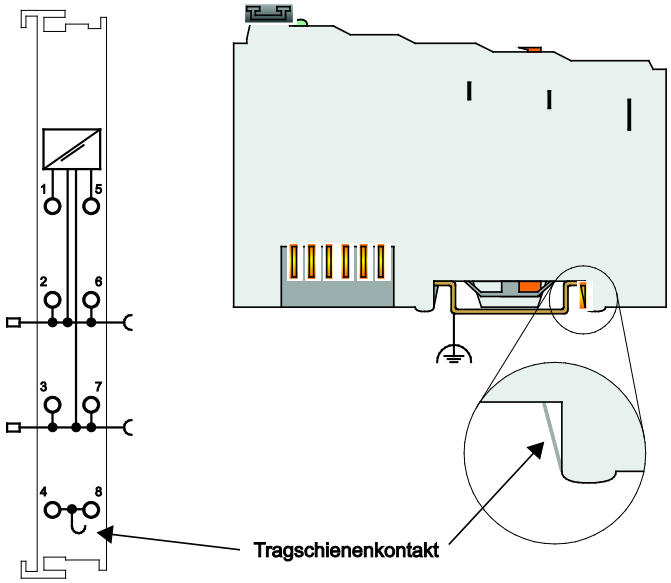


Abbildung 17: Tragschienenkontakt (Beispiel)

GEFAHR

Auf ausreichende Erdung achten!
Achten Sie auf den einwandfreien Kontakt zwischen dem Tragschienenkontakt und der Tragschiene. Die Tragschiene muss geerdet sein.
Beachten Sie dazu die Tragschieneneigenschaften, siehe Kapitel „Montage auf Tragschiene > Tragschieneneigenschaften“.

Die unteren CAGE CLAMP®-Anschlüsse von Einspeiseklemmen ermöglichen den optionalen Anschluss einer feldseitigen Funktionserde. Durch den unteren Federkontakt der drei Leistungskontakte wird dieses Potential der rechtsseitig angereihten Busklemme zur Verfügung gestellt. Einige Busklemmen verfügen über einen Messerkontakt, der dieses Potential abgreift. Dadurch wird hinsichtlich der Funktionserde eine Potentialgruppe mit der linksseitig angereihten Busklemme gebildet.

3.8 Schirmung

3.8.1 Allgemein

Der Einsatz geschirmter Kabel verringert elektromagnetische Einflüsse und erhöht damit die Signalqualität. Messfehler, Datenübertragungsfehler und Störung durch Überspannung können vermieden werden.

Hinweis

Kabelschirm mit Erdpotential verbinden!
Eine durchgängige Schirmung ist zwingend erforderlich, um die technischen Angaben bezüglich der Messgenauigkeit zu gewährleisten. Stellen Sie die Verbindung zwischen Kabelschirm und Erdpotential bereits am Einlass des Schrankes bzw. Gehäuses her. Dies ermöglicht es, eingestreute Störungen abzuleiten und von den darin befindlichen Geräten fern zu halten.

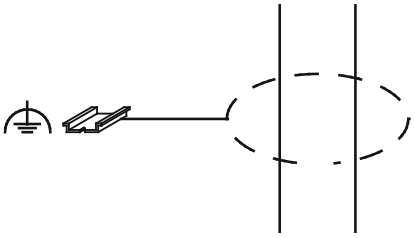


Abbildung 18: Kabelschirm auf Erdpotential

Hinweis

Verbessern der Schirmung durch großflächige Auflage!
Eine verbesserte Schirmung wird erreicht, wenn die Verbindung zwischen Schirm und Erdpotential niederohmig ist. Legen Sie zu diesem Zweck den Schirm großflächig auf, z. B. unter Verwendung des WAGO-Schirm-Anschlusssystems. Dies wird insbesondere für Anlagen mit großer Ausdehnung empfohlen, bei denen Ausgleichsströme fließen oder hohe impulsförmige Ströme (z. B. ausgelöst durch atmosphärische Entladung), auftreten können.

Hinweis

Daten- und Signalleitungen von Störquellen fernhalten!
Verlegen Sie Daten- und Signalleitungen getrennt von allen Starkstrom führenden Kabeln und anderen Quellen hoher elektromagnetischer Emission (z. B. Frequenzumrichter oder Antriebe).

3.8.2 Busleitungen

Die Schirmung der Busleitung ist in den jeweiligen Aufbaurichtlinien und Normen des Bussystems beschrieben.

3.8.3 Signalleitungen

Die Busklemmen für Analogsignale sowie einige Schnittstellen-Busklemmen besitzen Anschlussklemmen für den Schirm.

Hinweis

Geschirmte Signalleitungen verwenden!
Verwenden Sie für analoge Signale sowie an Busklemmen, welche über Anschlussklemmen für den Schirm verfügen, ausschließlich geschirmte Signalleitungen. Nur so ist gewährleistet, dass die für die jeweilige Busklemme angegebene Genauigkeit und Störfestigkeit auch bei Vorliegen von auf das Signalkabel einwirkenden Störungen erreicht werden.

3.8.4 WAGO-Schirm-Anschlusssystem

Das WAGO-Schirm-Anschlusssystem besteht aus Schirm-Klemmbügeln, Sammelschienen und diversen Montagefüßen. Mit diesen können viele verschiedene Aufbauten realisiert werden.

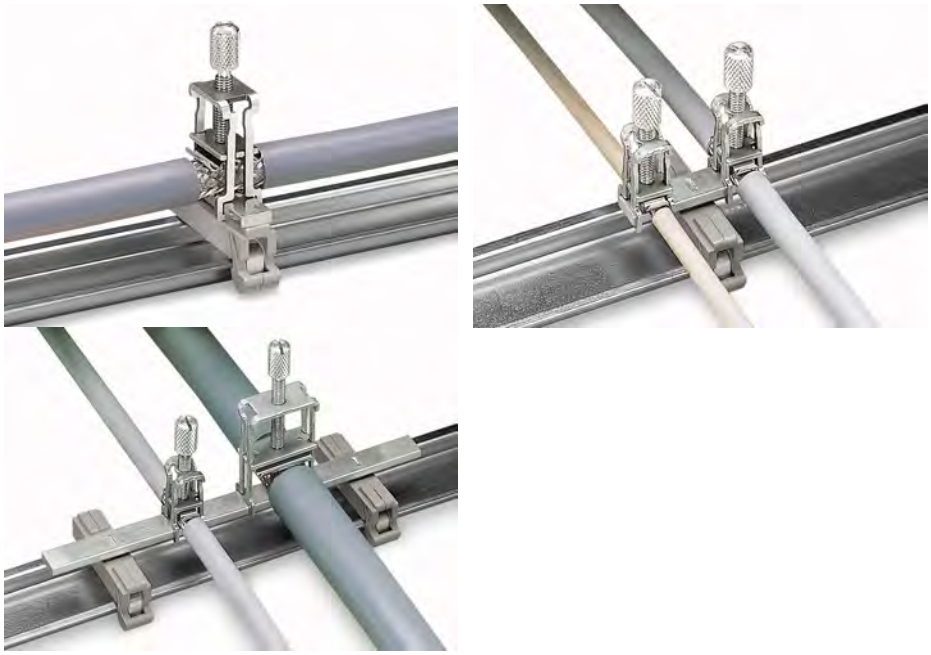


Abbildung 19: Beispiel WAGO-Schirm-Anschlusssystem

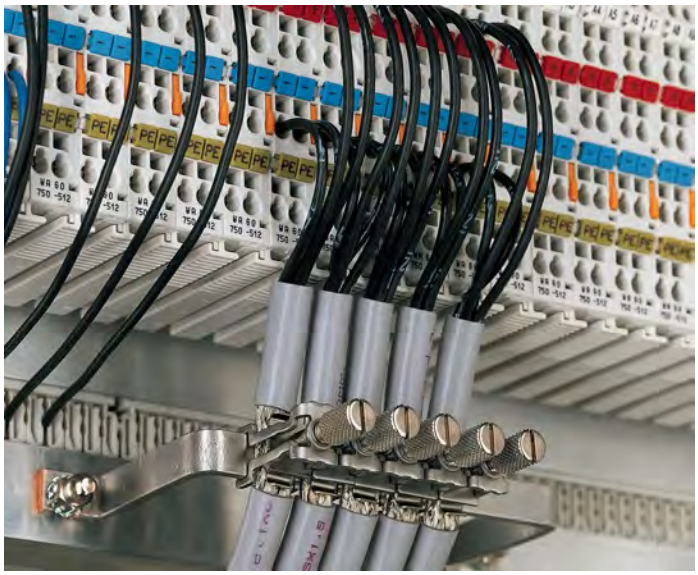


Abbildung 20: Anwendung des WAGO-Schirm-Anschlusssystems

4 Gerätebeschreibung

Der programmierbare Feldbuscontroller 750-852 (kurz: PFC) kombiniert die Funktionalität eines Feldbuskopplers zur Anschaltung an den Feldbus ETHERNET mit der einer Speicherprogrammierbaren Steuerung (SPS).

Dieser Feldbuscontroller kann für Anwendungen im Maschinen- und Anlagenbau sowie in der Prozessindustrie und der Gebäudetechnik eingesetzt werden.

Die zwei Ethernet-Schnittstellen und der integrierte Switch ermöglichen die Verdrahtung des Feldbusses in Linientopologie. Zusätzliche Infrastrukturelemente wie Switch oder Hub können somit entfallen. Beide Schnittstellen unterstützen Autonegotiation und Auto-MDI(X).

Mit dem DIP-Schalter kann das letzte Byte der IP-Adresse, sowie der Bezug der IP-Adresse (DHCP, BootP, feste Einstellung) vorgegeben werden.

In dem Feldbuscontroller werden sämtliche Eingangssignale der Sensoren zusammengeführt. Nach dem Einschalten des Feldbuscontrollers ermittelt der Feldbuscontroller alle in dem Knoten gesteckten Busklemmen und erstellt daraus ein lokales Prozessabbild. Hierbei kann es sich um eine gemischte Anordnung von analogen (Datenaustausch wortweise) und digitalen (Datenaustausch bitweise) Busklemmen handeln.

Das lokale Prozessabbild wird in einen Eingangs- und Ausgangsdatenbereich unterteilt.

Die Daten der analogen Busklemmen werden in der Reihenfolge ihrer Position nach dem Feldbuscontroller in das Prozessabbild gemappt.

Die Bits der digitalen Busklemmen werden zu Worten zusammengefügt und im Anschluss an die analogen ebenfalls in das Prozessabbild gemappt. Ist die Anzahl der digitalen Ein-/Ausgangsdaten größer als 16 Bit, beginnt der Feldbuscontroller automatisch ein weiteres Wort.

Entsprechend der IEC-61131-3-Programmierung erfolgt die Bearbeitung der Prozessdaten vor Ort in dem Feldbuscontroller. Die daraus erzeugten Verknüpfungsergebnisse können direkt an die Aktoren ausgegeben oder über den Bus an die übergeordnete Steuerung übertragen werden.

Die Feldbusanschaltung besteht aus zwei Ports (RJ-45). Ein im Feldbuscontroller integrierter ETHERNET-Switch, der im Store-And-Forward-Modus betrieben wird, verbindet die Feldbusports mit der CPU.
Beide Ports unterstützen:

- 10BASE-T / 100BASE-TX
- Voll- / Halbduplex
- Autonegotiation

- Auto-MDI(X)

Die Erstellung des Applikationsprogramms erfolgt mit WAGO-I/O-*PRO* gemäß IEC 61131-3, wobei die Basis von WAGO-I/O-*PRO* das Standard Programmiersystem CODESYS der Firma 3S ist, das mit den Target-Dateien für alle WAGO-Feldbuscontroller spezifisch erweitert wurde.

Für die IEC 61131-3 Programmierung stellt der Feldbuscontroller 512 KB Programmspeicher, 256 KB Datenspeicher und 8 KB Retain-Speicher zur Verfügung.

Der Anwender hat je nach Konfiguration Zugriff auf alle Feldbus- und Ein-/Ausgangsdaten.

Um Prozessdaten via ETHERNET zu versenden, unterstützt der Feldbuscontroller eine Reihe von Netzwerkprotokollen.

Für den Prozessdatenaustausch sind das MODBUS/TCP (UDP)-Protokoll und das ETHERNET/IP-Protokoll implementiert.
Beide Kommunikationsprotokolle können wahlweise oder parallel verwendet werden. Dazu wird die Schreibberechtigung auf die Busklemmen, d. h. der Zugriff auf diese von dem PFC aus über MODBUS/TCP oder EtherNet/IP, in einer XML-Datei festgelegt.

Als Konfigurations- und Diagnoseprotokolle stehen HTTP und SNTP zur Verfügung.

Zum Transfer von Daten über ETHERNET steht FTP zur Verfügung.

Zur automatischen Vergabe von IP-Adressen im Netzwerk kann wahlweise DHCP oder BootP genutzt werden.

Der Anwender kann Clients und Server über eine interne Socket-API für alle Transportprotokolle (TCP, UDP, u.s.w.) mit Funktionsbausteinen programmieren. Zur Funktionserweiterung sind Library-Funktionen verfügbar.

Der Feldbuscontroller basiert auf einer 32-Bit-CPU und ist multitasking-fähig, d. h. mehrere Programme können quasi-gleichzeitig ausgeführt werden.

Für die Konfiguration und Verwaltung des Systems bietet der Feldbuscontroller ein internes Dateisystem sowie einen integrierten Webserver.

Informationen über die Konfiguration und den Status des Feldbusknotens sind als HTML-Seiten in dem Feldbuscontroller gespeichert und können über einen Webbrowser ausgelesen werden. Darüber hinaus lassen sich über das implementierte Dateisystem auch eigene HTML-Seiten hinterlegen oder Programme direkt aufrufen.

Information

Kompatibilität mit der IEC-61131-3-Programmiersoftware!
Die Kompatibilität zwischen Ihrem Feldbuscontroller und der verwendeten IEC-61131-3-Programmiersoftware ist abhängig von der Firmware-Version des Feldbuscontrollers und der Version der Programmiersoftware.
Eine Referenzliste empfohlener Kombinationen finden Sie auf der Internetseite www.wago.com.
Verwenden Sie die Suchfunktion (Suchbegriff „Kompatibilitätshinweise“).

4.1 Ansicht

Die Ansicht zeigt die verschiedenen Einheiten des Gerätes:

- Auf der linken Seite befindet sich im unteren Bereich der Feldbusanschluss.
- Über dem Feldbusanschluss ist die Geräteeinspeisung mit Netzteil zur Systemversorgung.
- Auf der rechten Seite sind im oberen Bereich LEDs zur Statusanzeige der Buskommunikation, zur Fehlermeldung und Diagnose.
- Unten rechts ist an dem Gerät die Service-Schnittstelle zu finden.

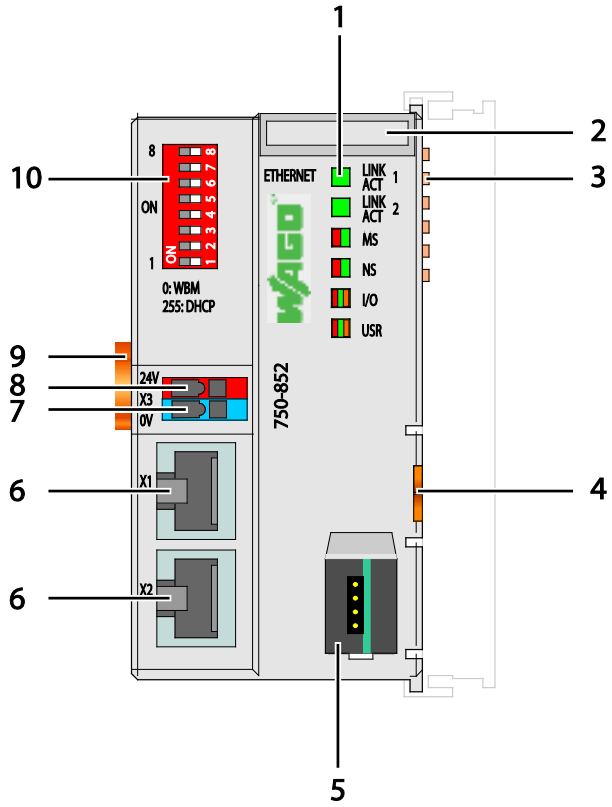


Abbildung 21: Ansicht ETHERNET TCP/IP-Feldbuscontroller

Tabelle 9: Legende zur Ansicht „ETHERNET-Feldbuscontroller“

Pos.	Be-zeichnung	Bedeutung	Details siehe Kapitel:
1	LINK ACT 1, 2 MS, NS, I/O, USR	Status-LEDs Feldbus	„Gerätebeschreibung“ > „Anzeigeelemente“
2	---	Befestigungsmöglichkeit für vier Mini-WSB-Schilder zur Beschriftung	---
3	---	Datenkontakte	„Geräte anschließen“ > „Klemmenbus/Daten-kontakte“
4	---	Entriegelungslasche	„Montieren“ > „Geräte einfügen und entfernen“
5	---	Service-Schnittstelle (Klappe geöffnet)	„Gerätebeschreibung“ > „Anschlüsse“
6	X1, X2	Feldbusanschluss 2 x RJ-45 als 2-Port-ETHERNET-Switch	„Gerätebeschreibung“ > „Anschlüsse“
7	-	CAGE CLAMP®–Anschlüsse Feldversorgung 0 V	„Systembeschreibung“ > „Spannungsversorgung“
8	+	CAGE CLAMP®–Anschlüsse Feldversorgung 24 V DC	„Systembeschreibung“ > „Spannungsversorgung“
9	---	Verriegelungsscheibe	„Montieren“ > „Geräte einfügen und entfernen“
10	---	Adresswahlschalter	„Gerätebeschreibung“ > „Bedienelemente“

4.2 Anschlüsse

4.2.1 Geräteeinspeisung

Die Versorgung wird über Klemmstellen mit CAGE CLAMP®-Anschluss eingespeist.

Das integrierte Netzteil erzeugt die erforderlichen Spannungen zur Versorgung der Elektronik und der angereihten Busklemmen.

Das Feldbus-Interface ist galvanisch von dem elektrischen Potential der Geräteelektronik getrennt.

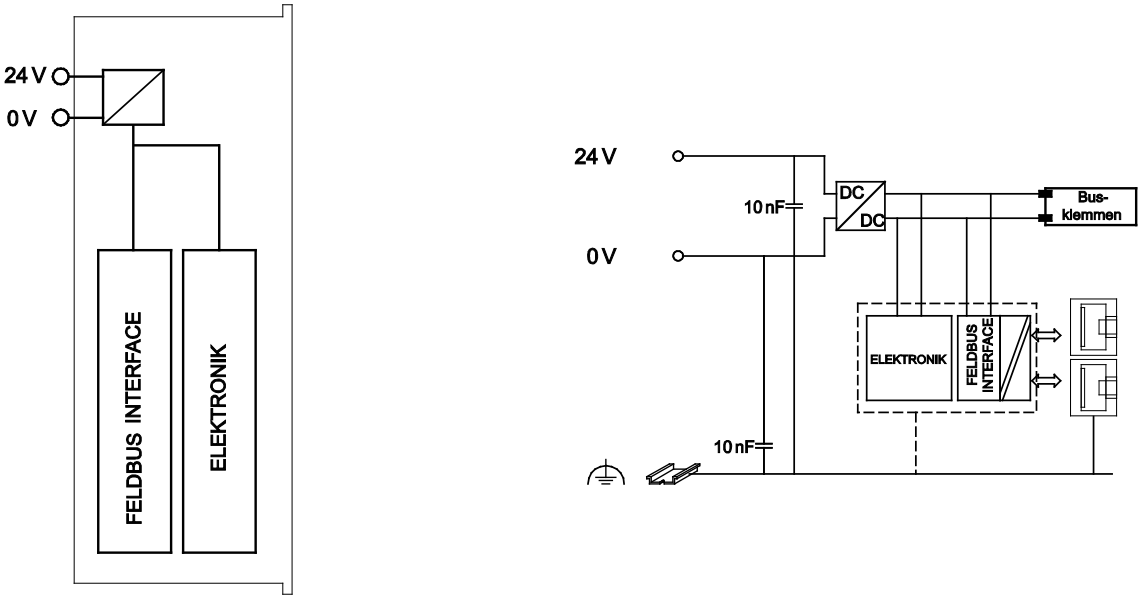


Abbildung 22: Geräteeinspeisung

4.2.2 Feldbusanschluss

Der Anschluss an die ETHERNET-basierenden Feldbusse erfolgt über zwei RJ-45-Steckverbinder, auch „Westernstecker“ genannt, die über einen integrierten Switch mit dem Feldbuscontroller verbunden sind.
Der integrierte Switch arbeitet im Store-and-Forward-Betrieb und unterstützt für jeden Port die Übertragungsgeschwindigkeiten 10/100 Mbit/s sowie die Übertragungsmodi Voll- bzw. Halbduplex.
Die RJ-45-Buchsen sind entsprechend den Vorgaben für 100BaseTX beschaltet. Als Verbindungsleitung wird vom ETHERNET-Standard ein Twisted-Pair-Kabel der Kategorie 5e vorgeschrieben. Dabei können Leitungen des Typs S-UTP (Screened-Unshielded Twisted Pair) sowie STP (Shielded Twisted Pair) mit einer maximalen Segmentlänge von 100 m benutzt werden.
Die Anschlussstelle ist so konzipiert, dass nach Steckeranschluss ein Einbau in einen 80 mm hohen Schaltkasten möglich ist.

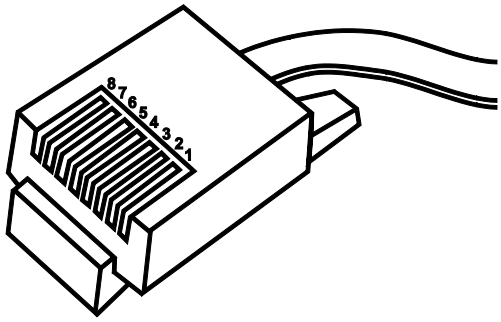


Abbildung 23: RJ-45-Stecker

Tabelle 10: Busanschluss und Steckerbelegung, RJ-45-Stecker

Kontakt	Signal	
1	TD +	Transmit Data +
2	TD -	Transmit Data -
3	RD +	Receive Data +
4		nicht belegt
5		nicht belegt
6	RD -	Receive Data -
7		nicht belegt
8		nicht belegt

ACHTUNG

Nicht in Telekommunikationsnetzen einsetzen!
Verwenden Sie Geräte mit ETHERNET-/RJ-45-Anschluss ausschließlich in LANs. Verbinden Sie diese Geräte niemals mit Telekommunikationsnetzen, wie z. B. mit Analog- oder ISDN-Telefonanlagen.

ACHTUNG

Nicht an einem PoE-Switch anschließen!
Schließen Sie den Feldbuskoppler nicht an einem PoE-Switch an.
Da der 8-polige „Power over Ethernet“-Anschluss mit Strom beschaltet ist, wird der Feldbuskoppler geschädigt.

4.3 Anzeigeelemente

Der Betriebszustand des Feldbuscontrollers bzw. des Knoten wird über Leuchtmelder in Form von Leuchtdioden (LEDs) signalisiert. Diese sind zum Teil mehrfarbig (rot, grün oder rot/grün (=orange)) ausgeführt.

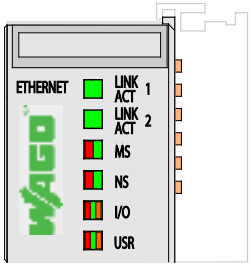


Abbildung 24: Anzeigeelemente

Zur Diagnose der verschiedenen Bereiche für Feldbus, Knoten und Versorgungsspannung werden entsprechend drei Gruppen von LEDs unterschieden:

Tabelle 11: Anzeigeelemente Feldbusstatus

LED	Farbe	Bedeutung
LINK ACT 1	grün	zeigt eine Verbindung zum physikalischen Netzwerk an Port 1 an
LINK ACT 2	grün	zeigt eine Verbindung zum physikalischen Netzwerk an Port 2 an
MS	rot/grün	zeigt den Knoten-Status an
NS	rot/grün	zeigt den Netzwerk-Status an

Tabelle 12: Anzeigeelemente Knotenstatus

LED	Farbe	Bedeutung
I/O	rot/grün/orange	zeigt den Klemmenbusbetrieb an und signalisiert Fehler mittels Blinkcodes
USR	rot/grün/orange	zeigt den, aus dem Anwenderprogramm heraus programmierten, Status an

Information

Weitere Informationen zu der LED-Signalisierung

Die detaillierte Beschreibung zur Auswertung der angezeigten LED-Zustände entnehmen Sie dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

4.4 Bedienelemente

4.4.1 Service-Schnittstelle

Die Service-Schnittstelle befindet sich hinter der Abdeckklappe.

Sie wird für die Kommunikation mit WAGO-I/O-*CHECK*, WAGO-I/O-*PRO* und zum Firmware-Update genutzt.

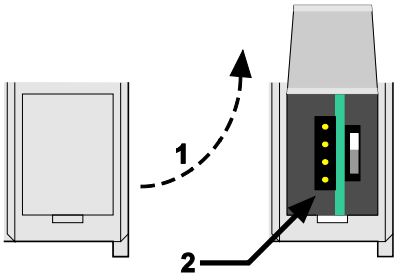


Abbildung 25: Service-Schnittstelle (geschlossene und geöffnete Klappe)

Tabelle 13: Legende zur Abbildung „Service-Schnittstelle (geschlossene und geöffnete Klappe)“

Nummer	Beschreibung
1	Geschlossene Klappe öffnen
2	Ansicht Service-Schnittstelle

ACHTUNG

Gerät muss spannungsfrei sein!
Um Geräteschäden zu vermeiden, ziehen und stecken Sie das Kommunikationskabel nur, wenn das Gerät spannungsfrei ist!

Der Anschluss an die 4-polige Stiftleiste unter der Abdeckklappe erfolgt über die Kommunikationskabel mit den Bestellnummern 750-920, 750-923 oder über den WAGO-Funkadapter mit der Bestellnummer 750-921.

4.4.2 Betriebsartenschalter

Der Betriebsartenschalter befindet sich hinter der Abdeckklappe.

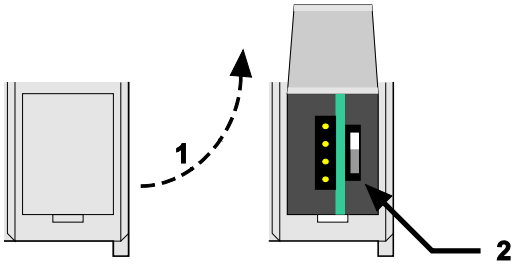


Abbildung 26: Betriebsartenschalter (geschlossene und geöffnete Klappe der Service-Schnittstelle)

Tabelle 14: Legende zur Abbildung „Betriebsartenschalter“

Nummer	Beschreibung
1	Geschlossene Klappe öffnen
2	Betriebsartenschalter

Der Betriebsartenschalter bestimmt das Laden, Starten und Stoppen der SPS-Applikation durch den Feldbuscontroller. Bei diesem Druck-/Schiebeschalter gibt es 3 Rast-Stellungen und eine Tastfunktion.
Der Schiebeschalter ist für eine Betätigungshäufigkeit nach EN 61131T2 ausgelegt.

ACHTUNG

Sachschäden durch gesetzte Ausgänge!
Beachten Sie, dass gesetzte Ausgänge weiterhin gesetzt bleiben, wenn Sie im laufenden Betrieb den Betriebsartenschalter von „RUN“ auf „STOP“ schalten! Da das Programm dann nicht mehr bearbeitet wird, sind softwareseitige Abschaltungen z. B. durch Initiatoren, unwirksam.
Programmieren bzw. definieren Sie deshalb alle Ausgänge, damit diese bei einem Programmstopp in einen sicheren Zustand schalten.

Hinweis

Definieren der Ausgänge für einen Programm-Stopp!
Um die Ausgänge bei Programm-Stopp in einen sicheren Zustand zu schalten, definieren Sie den Status der Ausgänge bei „STOP“.

1. Öffnen Sie dazu in dem Web-based Management-System (WBM) über den Link "PLC Settings" eine Web-Seite, auf der Sie die Funktion ***Process image - Set outputs to zero, if user program is stopped***, festlegen können.
2. Aktivieren Sie durch Setzen eines Häkchens in das Kontrollkästchen diese Funktion, werden alle Ausgänge auf Null gesetzt.
   Ist diese Funktion nicht aktiviert, verbleiben die Ausgänge auf dem letzten aktuellen Wert.

Hinweis

Bei Software-Start/-Stopp Betriebsartenschalterstellung unerheblich!
Die Stellung des Betriebsartenschalters ist für das Starten und Stoppen der PFC-Applikation aus WAGO-I/O-*PRO* heraus unerheblich.

Je nachdem, in welcher der drei statischen Positionen „Oben“, „Mitte“ oder „Unten“ sich der Schalter bei einem PowerOn oder einem Hard- oder Software-Reset befindet, ist eine der folgenden Funktionen aktiv:

Tabelle 15: Betriebsartenschalterstellungen, statische Positionen bei PowerOn/Reset

Stellung des Betriebsartenschalters	Funktion
Position „Oben“	„RUN“ – Programmbearbeitung aktivieren, Boot-Projekt (wenn vorhanden) wird gestartet
Position „Mitte“	„STOP“ – Programmbearbeitung stoppen, PFC-Applikation wird angehalten
Position „Unten“	Nicht verwenden. Diese Position ist für den Anwender nicht relevant.

Wird während des laufenden Betriebs ein Stellungswechsel des Schalters vorgenommen, führt der Feldbuscontroller die folgenden Funktionen aus:

Tabelle 16: Betriebsartenschalterstellungen, dynamische Positionen im laufenden Betrieb

Stellungswechsel des Betriebsartenschalters	Funktion
Von obere in mittlere Position	„STOP“ – Programmbearbeitung stoppen, PFC-Applikation wird angehalten.
Von mittlere in obere Position	„RUN“ – Programmbearbeitung aktivieren, Boot-Projekt (wenn vorhanden) wird gestartet.
Von mittlere in untere Position	Es erfolgt keine Reaktion. Nach PowerOn/Reset wird der Bootstrap-Loader gestartet.
Von untere in mittlere Position	Es erfolgt keine Reaktion.
Niederdrücken (z. B. mit Schraubendreher)	Hardware-Reset. Alle Ausgänge werden zurückgesetzt; Variablen werden auf 0 bzw. auf FALSE oder auf einen Initialwert gesetzt. Retain-Variablen, bzw. Merker werden nicht verändert. Der Hardware-Reset kann sowohl bei STOP als auch bei RUN in jeder Stellung des Betriebsartenschalters ausgeführt werden! Neuanlauf des Feldbuscontrollers.

Der Wechsel der Betriebsart erfolgt intern am Ende eines PFC-Zyklus.

4.4.3 Adresswahlschalter

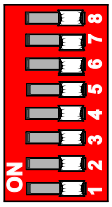


Abbildung 27: Adresswahlschalter (hier Einstellung „0“)

Der 8-polige DIP-Schalter dient zum Einstellen der IP-Adresse und zur Auswahl des Protokolls, mit dem die IP-Adresse eingestellt werden kann.

Die IP-Adresse setzt sich aus einem Netzanteil und einem Hostanteil zusammen.

Netzanteil			Hostanteil
192.	168.	1.	0…255

Der Netzanteil ist konfigurierbar und im Auslieferungszustand 192.168.1.
Der Hostanteil wird über den DIP-Schalter eingestellt.

Die einzelnen DIP-Schalter 1 bis 8 repräsentieren eine Bitmaske entsprechend den Werten 0 bis 255. Dabei entspricht Schalter 1 dem niederwertigsten Bit (2^0) und Schalter 8 dem höchstwertigsten Bit (2^7).

Tabelle 17: Bedeutung Schalterstellungen des DIP-Schalters

Hostanteil der IP-Adresse	**Bedeutung**
0 (WBM)	Es wird die IP-Konfiguration verwendet, die im Web-based Management (WBM) auf der Seite „Port“ eingestellt ist (Standardmäßig BootP, jedoch auf DHCP oder EEPROM änderbar).
1-254	Hostanteil nach Schalterstellung Beispiel für die Einstellung des DIP-Schalters: Für die IP-Adresse 192.168.1.**33** sind der 1. und der 6. Schalter zu setzen: 2^0 (1. Schalter) + 2^5 (6. Schalter) = 33
255 (DHCP)	Die Konfiguration der IP-Parameter wird mittels DHCP-Protokoll vorgenommen.

4.5 Technische Daten

4.5.1 Gerätedaten

Tabelle 18: Technische Daten – Gerätedaten

Breite	50 mm
Höhe (ab Oberkante Tragschiene)	65 mm (ab Oberkante Tragschiene)
Tiefe	97 mm
Gewicht	112 g
Schutzart	IP 20

4.5.2 Systemdaten

Tabelle 19: Technische Daten – Systemdaten

Max. Anzahl Busteilnehmer	Limitiert durch ETHERNET-Spezifikation
Übertragungsmedium	Twisted Pair S/UTP, STP 100 Ω Cat 5
Busanschluss	2 x RJ-45
Bussegmentlänge $_{max}$	100 m nach Hub und 750-852
Netzwerklänge $_{max}$	2000 m
Übertragungsrate	10/100 Mbit/s
Protokolle	MODBUS/TCP (UDP), EtherNet/IP, HTTP, BootP, DHCP, DNS, SNTP, FTP
Programmierung	WAGO-I/O-*PRO*
IEC-61131-3	AWL, KOP, FUP (CFC), ST, AS
Max. Anzahl Socket-Verbindungen	3 HTTP, 15 MODBUS/TCP, 10 FTP, 5 für IEC-61131-3-Programme, 2 für WAGO-I/O-*PRO*, 128 für Ethernet/IP
Anzahl Busklemmen - mit Busverlängerung	64 250
Konfiguration	über PC
Programmspeicher	512 kByte
Datenspeicher	256 kByte
Remanentspeicher	8 kByte (4 kByte Retain, 4 kByte Merker)

4.5.3 Versorgung

Tabelle 20: Technische Daten – Versorgung

Spannungsversorgung	DC 24 V (-25 % ... +30 %)
Eingangstrom $_{max.}$	300 mA bei 24 V
Netzteilwirkungsgrad	90 %
Interne Stromaufnahme	400 mA bei 5 V
Summenstrom für Busklemmen	700 mA bei 5 V
Potentialtrennung	500 V System/Versorgung

4.5.4 Feldbus MODBUS/TCP

Tabelle 21: Technische Daten – Feldbus MODBUS/TCP

Eingangsprozessabbild $_{max}$	2040 Byte
Ausgangsprozessabbild $_{max}$	2040 Byte
Eingangsvariablen $_{max}$	512 Byte
Ausgangsvariablen $_{max}$	512 Byte

4.5.5 Zubehör

Tabelle 22: Technische Daten – Zubehör

Mini-WSB-Schnellbezeichnungssystem	
WAGO-I/O-*PRO*	

4.5.6 Anschlusstechnik

Tabelle 23: Technische Daten – Verdrahtungsebene

Anschlusstechnik	Push-in CAGE CLAMP®
Leiterquerschnitt	0,08 mm² … 1,5 mm² / AWG 28 … 16
Abisolierlänge	8 mm … 9 mm / 0,33 in

Tabelle 24: Technische Daten – Leistungskontakte

Leistungskontakte	Messer-/Federkontakt, selbstreinigend
Spannungsabfall bei I $_{max.}$	< 1 V bei 64 Busklemmen

Tabelle 25: Technische Daten – Datenkontakte

Datenkontakte	Gleitkontakte, hartvergoldet, selbstreinigend

4.5.7 Klimatische Umweltbedingungen

Tabelle 26: Technische Daten – klimatische Umweltbedingungen

Betriebstemperaturbereich	0 °C … 55 °C
Lagertemperaturbereich	−25 °C … +85 °C
Relative Feuchte (ohne Betauung)	max. 95 %
Beanspruchung durch Schadstoffe	gem. IEC 60068-2-42 und IEC 60068-2-43
Max. Schadstoffkonzentration bei einer relativen Feuchte < 75 %	$SO_2 \leq 25$ ppm $H_2S \leq 10$ ppm
Besondere Bedingungen	Die Komponenten dürfen nicht ohne Zusatzmaßnahmen an Orten eingesetzt werden, an denen Staub, ätzende Dämpfe, Gase oder ionisierende Strahlung auftreten können.

4.5.8 Mechanische Festigkeit

Tabelle 27: Technische Daten – Mechanische Festigkeit

Vibrationsfestigkeit	gem. IEC 60068-2-6 Anmerkung zur Schwingungsprüfung: a) Schwingungsart: Frequenzdurchläufe mit einer Änderungsgeschwindigkeit von 1 Oktave/Minute 10 Hz ≤ f < 57 Hz, Amplitude 0,075 mm konstant , 57 Hz ≤ f ≤ 150 Hz, konstant Beschleunigung: 1 g b) Schwingungsdauer: 10 Frequenzdurchläufe pro Achse in jeder der 3 zueinander senkrechten Achsen
Schockfestigkeit	gem. IEC 60068-2-27 Anmerkung zur Stoßprüfung: a) Art des Stoßes: Halbsinus b) Stoßstärke: 15 g Scheitelwert, 11 ms Dauer c) Stoßrichtung: je 3 Stöße in pos. und neg. Richtung der 3 senkrecht zueinanderstehenden Achsen des Prüflings, also insgesamt 18 Schocks.
Freier Fall	gem. IEC 60068-2-32 ≤ 1 m (Gerät in Originalverpackung)

4.6 Zulassungen

Information **Weitere Informationen zu Zulassungen**

Detaillierte Hinweise zu den Zulassungen können Sie dem Dokument „Übersicht Zulassungen **WAGO-I/O-SYSTEM 750**“ entnehmen. Dieses finden Sie im Internet unter: www.wago.com → Service → Downloads → Zusätzliche Dokumentation und Information für Automatisierungsprodukte → WAGO-I/O-SYSTEM 750 →Systembeschreibung.

Folgende Zulassungen wurden für den Feldbuskoppler/-controller 750-852 erteilt:

CE	Konformitätskennzeichnung	
cULus	$_{C}UL_{US}$	UL508

Folgende Ex-Zulassungen wurden für den Feldbuskoppler/-controller 750-852 erteilt:

cULus	$_{C}UL_{US}$	ANSI/ISA 12.12.01
	Class I, Div2 ABCD T4	

Folgende Schiffszulassungen wurden für den Feldbuskoppler/-controller 750-852 erteilt:

GL	GL (Germanischer Lloyd)	Cat. A, B, C, D (EMC 1)

Information **Weitere Information zu den Schiffszulassungen**

Beachten Sie zu den Schiffszulassungen das Kapitel „Ergänzende Einspeisevorschriften“.

4.7 Normen und Richtlinien

Der Feldbuskoppler/-controller 750-852 erfüllt folgende EMV-Normen:

EMV CE-Störfestigkeit	gem. EN 61000-6-2
EMV CE-Störaussendung	gem. EN 61000-6-3
EMV Schiffbau-Störfestigkeit	gem. Germanischer Lloyd
EMV Schiffbau-Störaussendung	gem. Germanischer Lloyd

5 Montieren

5.1 Einbaulage

Neben dem horizontalen und vertikalen Einbau sind alle anderen Einbaulagen erlaubt.

Hinweis

Bei vertikalem Einbau Endklammer verwenden!
Montieren Sie beim vertikalen Einbau zusätzlich unterhalb des Feldbusknotens eine Endklammer, um den Feldbusknoten gegen Abrutschen zu sichern.

WAGO-Bestellnummer 249-116	Endklammer für TS 35, 6 mm breit
WAGO-Bestellnummer 249-117	Endklammer für TS 35, 10 mm breit

5.2 Gesamtaufbau

Die maximale Gesamtausdehnung eines Feldbusknotens ohne Feldbuskoppler/-controller beträgt 780 mm inklusive Endklemme. Die Breite der Endklemme beträgt 12 mm. Die übrigen Busklemmen verteilen sich also auf einer Länge von maximal 768 mm.

Beispiele:

- An einen Feldbuskoppler/-controller können 64 Ein- und Ausgangsbusklemmen der Breite 12 mm gesteckt werden.
- An einen Feldbuskoppler/-controller können 32 Ein- und Ausgangsbusklemmen der Breite 24 mm gesteckt werden.

Ausnahme:

Die Anzahl der gesteckten Busklemmen hängt außerdem vom jeweiligen Feldbuskoppler/-controller ab, an dem sie betrieben werden. Beispielsweise beträgt die maximale Anzahl der anreihbaren Busklemmen an einem PROFIBUS-DP/V1-Feldbuskoppler/-controller 63 Busklemmen ohne passive Busklemmen und Endklemme.

ACHTUNG

Maximale Gesamtausdehnung eines Feldbusknotens beachten!
Die maximale Gesamtausdehnung eines Feldbusknotens ohne Feldbuskoppler/-controller und ohne die Nutzung einer Busklemme 750-628 (Kopplerklemme zur Klemmenbusverlängerung) darf eine Länge von 780 mm nicht überschreiten.
Beachten Sie zudem Einschränkungen einzelner Feldbuskoppler/-controller.

Hinweis

Gesamtausdehnung mit Kopplerklemme zur Klemmenbusverlängerung erhöhen!

Mit der Busklemme 750-628 (Kopplerklemme zur Klemmenbusverlängerung) können Sie die Gesamtausdehnung eines Feldbusknotens erhöhen. Bei einem solchen Aufbau stecken Sie nach der letzten Busklemme eines Klemmenblocks eine Busklemme 750-627 (Endklemme zur Klemmenbusverlängerung. Diese verbinden Sie per RJ-45-Patch-Kabel mit der Kopplerklemme zur Klemmenbusverlängerung eines weiteren Klemmenblocks.

So können Sie mit maximal 10 Busklemmen zur Klemmenbusverlängerung einen Feldbusknoten mechanisch in maximal 11 Blöcke aufteilen.

Die zulässige Kabellänge zwischen zwei Blöcken beträgt 5 Meter.

Weitere Informationen finden Sie in den Handbüchern der Busklemmen 750-627 und 750-628).

5.3 Montage auf Tragschiene

5.3.1 Tragschieneneigenschaften

Alle Komponenten des Systems können direkt auf eine Tragschiene gemäß EN 50022 (TS 35, DIN Rail 35) aufgerastet werden.

ACHTUNG

Ohne Freigabe keine WAGO-fremden Tragschienen verwenden!
WAGO liefert normkonforme Tragschienen, die optimal für den Einsatz mit dem WAGO-I/O-SYSTEM geeignet sind. Sollten Sie andere Tragschienen einsetzen, muss eine technische Untersuchung und eine Freigabe durch WAGO Kontakttechnik GmbH & Co. KG vorgenommen werden.

Tragschienen weisen unterschiedliche mechanische und elektrische Merkmale auf. Für den optimalen Aufbau des Systems auf einer Tragschiene sind Randbedingungen zu beachten:

- Das Material muss korrosionsbeständig sein.
- Die meisten Komponenten besitzen zur Ableitung von elektromagnetischen Einflüssen einen Ableitkontakt zur Tragschiene. Um Korrosionseinflüssen vorzubeugen, darf dieser verzinnte Tragschienenkontakt mit dem Material der Tragschiene kein galvanisches Element bilden, das eine Differenzspannung über 0,5 V (Kochsalzlösung von 0,3 % bei 20 °C) erzeugt.
- Die Tragschiene muss die im System integrierten EMV-Maßnahmen und die Schirmung über die Busklemmenanschlüsse optimal unterstützen.
- Eine ausreichend stabile Tragschiene ist auszuwählen und ggf. mehrere Montagepunkte (alle 20 cm) für die Tragschiene zu nutzen, um Durchbiegen und Verdrehung (Torsion) zu verhindern.
- Die Geometrie der Tragschiene darf nicht verändert werden, um den sicheren Halt der Komponenten sicherzustellen. Insbesondere beim Kürzen und Montieren darf die Tragschiene nicht gequetscht oder gebogen werden.
- Der Rastfuß der Komponenten reicht in das Profil der Tragschiene hinein. Bei Tragschienen mit einer Höhe von 7,5 mm sind Montagepunkte (Verschraubungen) unter dem Knoten in der Tragschiene zu versenken (Senkkopfschrauben oder Blindnieten).
- Die Metallfedern auf der Gehäuseunterseite müssen einen niederimpedanten Kontakt zur Tragschiene haben (möglichst breitflächige Auflage).

5.3.2 WAGO-Tragschienen

Die WAGO-Tragschienen erfüllen die elektrischen und mechanischen Anforderungen.

Tabelle 28: WAGO-Tragschienen

Bestellnummer	Beschreibung
210-113 /-112	35 x 7,5; 1 mm Stahl gelb chromatiert; gelocht/ungelocht
210-114 /-197	35 x 15; 1,5 mm Stahl gelb chromatiert; gelocht/ungelocht
210-118	35 x 15; 2,3 mm Stahl gelb chromatiert; ungelocht
210-198	35 x 15; 2,3 mm Kupfer; ungelocht
210-196	35 x 7,5; 1 mm Aluminium; ungelocht

5.4 Abstände

Für den gesamten Feldbusknoten sind Abstände zu benachbarten Komponenten, Kabelkanälen und Gehäuse-/Rahmenwänden einzuhalten.

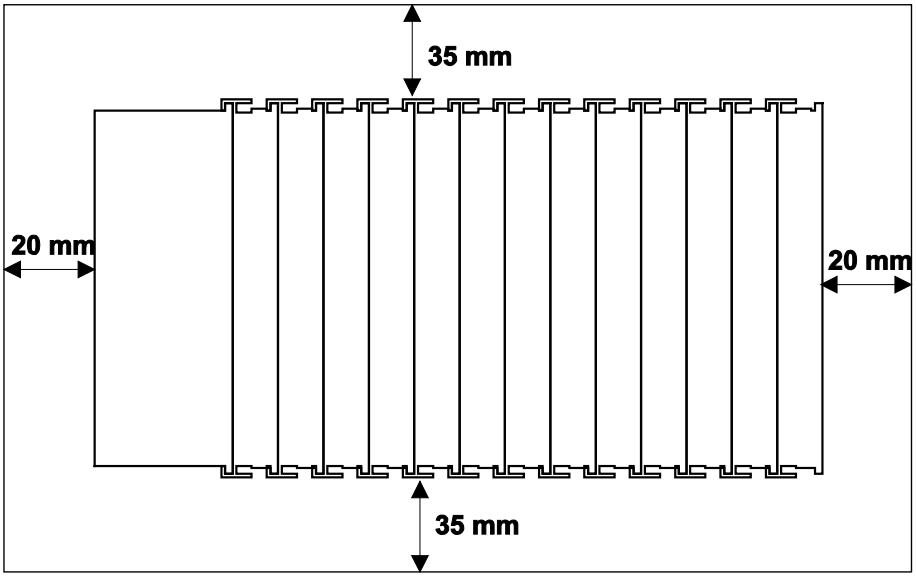


Abbildung 28: Abstände

Die Abstände schaffen Raum zur Wärmeableitung und Montage bzw. Verdrahtung. Ebenso verhindern die Abstände zu Kabelkanälen, dass leitungsgebundene elektromagnetische Störungen den Betrieb beeinflussen.

5.5 Montagereihenfolge

Feldbuskoppler/-controller und Busklemmen des WAGO-I/O-SYSTEMs 750/753 werden direkt auf eine Tragschiene gemäß EN 50022 (TS 35) aufgerastet.

Die sichere Positionierung und Verbindung erfolgt über ein Nut- und Feder-System. Eine automatische Verriegelung garantiert den sicheren Halt auf der Tragschiene.

Beginnend mit dem Feldbuskoppler/-controller werden die Busklemmen entsprechend der Projektierung aneinandergereiht. Fehler bei der Projektierung des Knotens bezüglich der Potentialgruppen (Verbindungen über die Leistungskontakte) werden erkannt, da Busklemmen mit Leistungskontakten (Messerkontakte) nicht an Busklemmen angereiht werden können, die weniger Leistungskontakte besitzen.

VORSICHT

Verletzungsgefahr durch scharfkantige Messerkontakte!
Da die Messerkontakte sehr scharfkantig sind, besteht bei unvorsichtiger Hantierung mit den Busklemmen Verletzungsgefahr.

ACHTUNG

Busklemmen nur in vorgesehener Reihenfolge stecken!
Alle Busklemmen verfügen an der rechten Seite über Nuten zur Aufnahme von Messerkontakten. Bei einigen Busklemmen sind die Nuten oben verschlossen. Andere Busklemmen, die an dieser Stelle linksseitig über einen Messerkontakt verfügen, können dann nicht von oben angesteckt werden. Diese mechanische Kodierung hilft dabei, Projektierungsfehler zu vermeiden, die zur Zerstörung der Komponenten führen können. Stecken Sie Busklemmen daher ausschließlich von rechts und von oben.

Hinweis

Busabschluss nicht vergessen!
Stecken Sie immer eine Busendklemme 750-600 an das Ende des Feldbusknotens! Die Busendklemme muss in allen Feldbusknoten mit Feldbuskopplern/-controllern des WAGO-I/O-SYSTEMs 750 eingesetzt werden, um eine ordnungsgemäße Datenübertragung zu garantieren!

5.6 Geräte einfügen und entfernen

ACHTUNG

Arbeiten an Geräten nur spannungsfrei durchführen!
Arbeiten unter Spannung können zu Schäden an den Geräten führen. Schalten Sie daher die Spannungsversorgung ab, bevor Sie an den Geräten arbeiten.

5.6.1 Feldbuskoppler/-controller einfügen

1. Wenn Sie den Feldbuskoppler/-controller gegen einen bereits vorhandenen Feldbuskoppler/-controller austauschen, positionieren Sie den neuen Feldbuskoppler/-controller so, dass Nut und Feder zur nachfolgenden Busklemme verbunden sind.

2. Rasten Sie den Feldbuskoppler/-controller auf die Tragschiene auf.

3. Drehen Sie die Verriegelungsscheibe mit einer Schraubendreherklinge, bis die Nase der Verriegelungsscheibe hinter der Tragschiene einrastet (siehe nachfolgende Abbildung). Damit ist der Feldbuskoppler/-controller auf der Tragschiene gegen Verkanten gesichert.

Mit dem Einrasten des Feldbuskopplers/-controllers sind die elektrischen Verbindungen der Datenkontakte und (soweit vorhanden) der Leistungskontakte zur gegebenenfalls nachfolgenden Busklemme hergestellt.

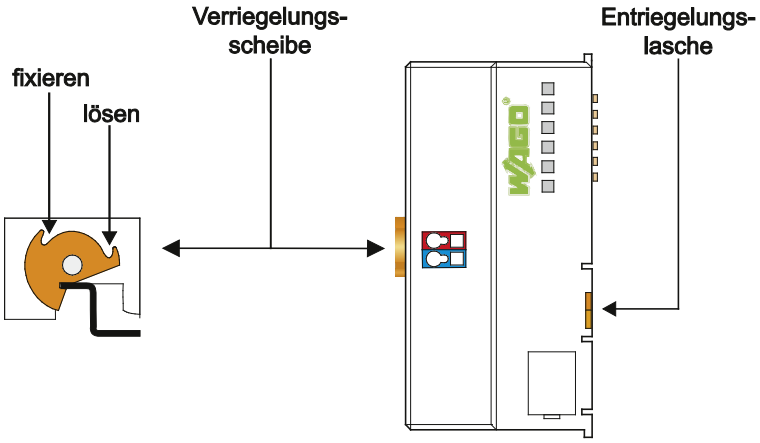


Abbildung 29: Verriegelung

5.6.2 Feldbuskoppler/-controller entfernen

1. Drehen Sie die Verriegelungsscheibe mit einer Schraubendreherklinge, bis die Nase der Verriegelungsscheibe nicht mehr hinter der Tragschiene eingerastet ist.

2. Ziehen Sie den Feldbuskoppler/-controller an der Entriegelungslasche aus dem Verbund.

Mit dem Herausziehen des Feldbuskopplers/-controllers sind die elektrischen Verbindungen der Datenkontakte bzw. Leistungskontakte zu nachfolgenden Busklemmen wieder getrennt.

5.6.3 Busklemme einfügen

1. Positionieren Sie die Busklemme so, dass Nut und Feder zum Feldbuskoppler/-controller oder zur vorhergehenden und gegebenenfalls zur nachfolgenden Busklemme verbunden sind.

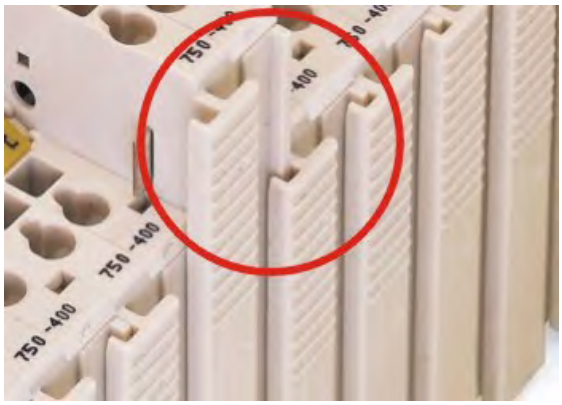


Abbildung 30: Busklemme einsetzen (Beispiel)

2. Drücken Sie die Busklemme in den Verbund, bis die Busklemme auf der Tragschiene einrastet.

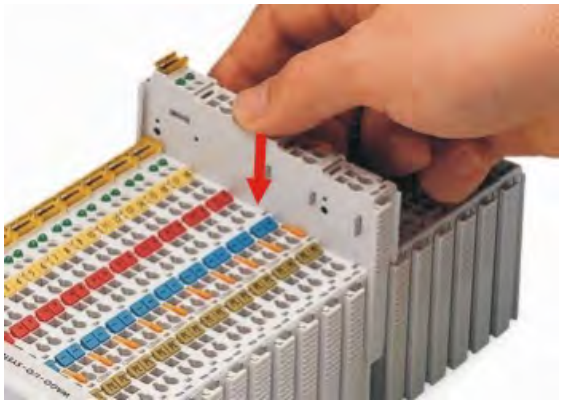


Abbildung 31: Busklemme einrasten (Beispiel)

Mit dem Einrasten der Busklemme sind die elektrischen Verbindungen der Datenkontakte und (soweit vorhanden) der Leistungskontakte zum Feldbuskoppler/-controller oder zur vorhergehenden und gegebenenfalls zur nachfolgenden Busklemme hergestellt.

5.6.4 Busklemme entfernen

1. Ziehen Sie die Busklemme an der Entriegelungslasche aus dem Verbund.

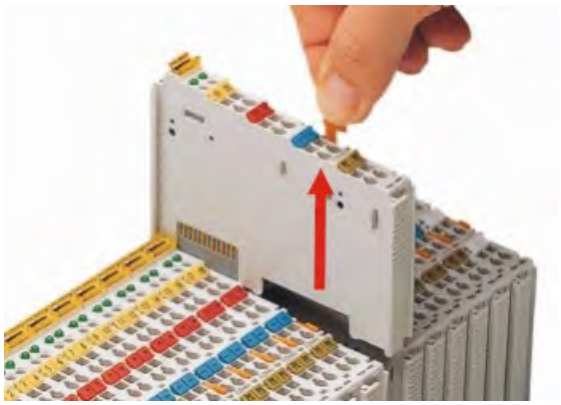


Abbildung 32: Busklemme entfernen (Beispiel)

Mit dem Herausziehen der Busklemme sind die elektrischen Verbindungen der Datenkontakte bzw. Leistungskontakte wieder getrennt.

6 Geräte anschließen

6.1 Datenkontakte/Klemmenbus

Die Kommunikation zwischen Feldbuskoppler/-controller und Busklemmen sowie die Systemversorgung der Busklemmen erfolgt über den Klemmenbus. Er besteht aus 6 Datenkontakten, die als selbstreinigende Goldfederkontakte ausgeführt sind.

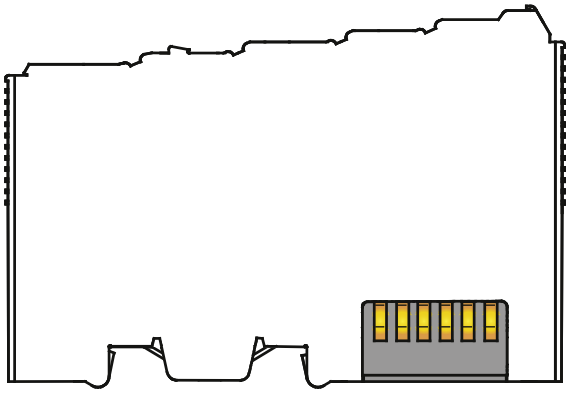


Abbildung 33: Datenkontakte

ACHTUNG

Busklemmen nicht auf Goldfederkontakte legen!
Um Verschmutzung und Kratzer zu vermeiden, legen Sie die Busklemmen nicht auf die Goldfederkontakte.

ESD

Auf gute Erdung der Umgebung achten!
Die Geräte sind mit elektronischen Bauelementen bestückt, die bei elektrostatischer Entladung zerstört werden können. Achten Sie beim Umgang mit den Geräten auf gute Erdung der Umgebung (Personen, Arbeitsplatz und Verpackung). Berühren Sie keine elektrisch leitenden Bauteile, z. B. Datenkontakte.

6.2 Leistungskontakte/Feldversorgung

VORSICHT

Verletzungsgefahr durch scharfkantige Messerkontakte!
Da die Messerkontakte sehr scharfkantig sind, besteht bei unvorsichtiger Hantierung mit den Busklemmen Verletzungsgefahr.

Auf der rechten Seite der meisten Feldbuskoppler/-controller und einiger Busklemmen befinden sich selbstreinigende Leistungskontakte.
Die Leistungskontakte leiten die Versorgungsspannung für die Feldseite weiter.
Die Kontakte sind berührungssicher als Federkontakte ausgeführt.
Als Gegenstück sind auf der linken Seite der Busklemmen entsprechende Messerkontakte vorhanden.

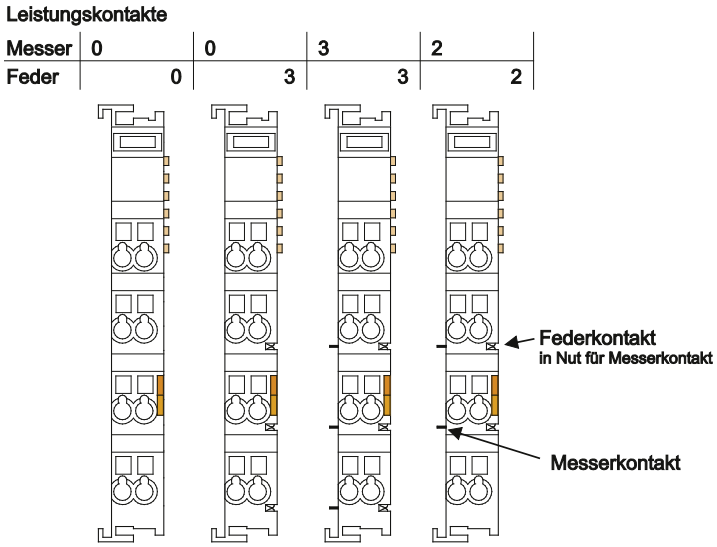


Abbildung 34: Beispiele für die Anordnung von Leistungskontakten

Hinweis

Feldbusknoten mit smartDESIGNER konfigurieren und überprüfen
Sie können mit der WAGO-ProServe®-Software **smartDESIGNER** den Aufbau eines Feldbusknotens konfigurieren. Über die integrierte Plausibilitätsprüfung können Sie die Konfiguration überprüfen.

6.3 Leiter an CAGE CLAMP® anschließen

CAGE CLAMP®-Anschlüsse von WAGO sind für ein-, mehr- oder feindrahtige Leiter ausgelegt.

Hinweis

Nur einen Leiter pro CAGE CLAMP® anschließen!
Sie dürfen an jedem CAGE CLAMP®-Anschluss nur einen Leiter anschließen. Mehrere einzelne Leiter an einem Anschluss sind nicht zulässig.

Müssen mehrere Leiter auf einen Anschluss gelegt werden, verbinden Sie diese in einer vorgelagerten Verdrahtung, z. B. mit WAGO-Durchgangsklemmen.

Ausnahme:
Sollte es unvermeidbar sein, zwei mehr- oder feindrahtige Leiter an einem CAGE CLAMP®-Anschluss anzuschließen, müssen Sie eine gemeinsame Aderendhülse verwenden. Folgende Aderendhülsen sind einsetzbar:

Länge	8 mm
Nennquerschnitt $_{max.}$	1 mm² für zwei mehr- oder feindrahtige Leiter mit je 0,5 mm²
WAGO-Produkt	216-103 oder Produkte mit gleichen Eigenschaften.

1. Zum Öffnen der CAGE CLAMP® führen Sie das Betätigungswerkzeug in die Öffnung oberhalb des Anschlusses ein.

2. Führen Sie den Leiter in die entsprechende Anschlussöffnung ein.

3. Zum Schließen der CAGE CLAMP® entfernen Sie das Betätigungswerkzeug wieder. Der Leiter ist festgeklemmt.

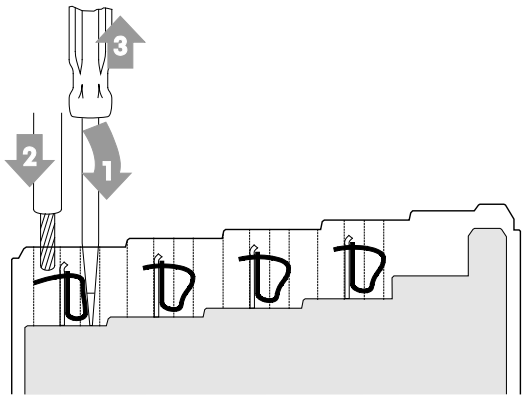


Abbildung 35: Leiter an CAGE CLAMP® anschließen

7 Funktionsbeschreibung

7.1 Betriebssystem

7.1.1 Anlauf des Feldbuscontrollers

Hinweis

Betriebsartenschalter darf sich nicht in der unteren Stellung befinden!
Damit ein Anlauf erfolgen kann, darf der Betriebsartenschalter beim Anlauf nicht in die untere Stellung geschaltet sein!

Nach Einschalten der Versorgungsspannung oder nach Hardware-Reset startet der Feldbuscontroller.

Das intern vorhandene PFC-Programm wird ins RAM übertragen.

In der Initialisierungsphase ermittelt der Feldbuscontroller die Busklemmen und die vorliegende Konfiguration und setzt die Variablen auf 0 bzw. auf FALSE oder auf einen von dem PFC-Programm vorgegebenen Initialwert.
Die Merker behalten ihren Zustand bei.
Während dieser Phase blinkt die I/O-LED rot.

Nach fehlerfreiem Anlauf leuchtet die I/O-LED grün.

Information

Weitere Informationen zu der LED-Signalisierung
Die detaillierte Beschreibung zur Auswertung der angezeigten LED-Zustände entnehmen Sie dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

7.1.2 PFC-Zyklus

Nach fehlerfreiem Anlauf startet der PFC-Zyklus bei oberer Stellung des Betriebsartenschalters oder durch einen Start-Befehl aus WAGO-I/O-*PRO*.
Die Ein- und Ausgangsdaten des Feldbusses und der Busklemmen sowie die Werte von Zeitgebern werden gelesen. Anschließend wird das im RAM vorhandene PFC-Programm bearbeitet und danach die Ausgangsdaten des Feldbusses und der Busklemmen ins Prozessabbild geschrieben. Am Ende des PFC-Zyklus werden Betriebssystemfunktionen u. a. für Diagnose und Kommunikation ausgeführt und die Werte von Zeitgebern aktualisiert.
Der Zyklus beginnt erneut mit dem Einlesen der Ein- und Ausgangsdaten und der Werte von Zeitgebern.

Der Wechsel der Betriebsart (STOP/RUN) erfolgt am Ende eines PFC-Zyklus.

Die Zykluszeit ist die Zeit vom Beginn des PFC-Programms bis zum nächsten Beginn. Wenn innerhalb eines PFC-Programms eine Schleife programmiert wird, verlängert sich entsprechend die PFC-Laufzeit und somit der PFC-Zyklus.

Während der Bearbeitung des PFC-Programms werden die Eingänge, Ausgänge und Werte von Zeitgebern nicht aktualisiert. Diese Aktualisierung findet erst definiert am Ende des PFC-Programms statt. Hieraus ergibt sich, dass es nicht möglich ist, innerhalb einer Schleife auf ein Ereignis aus dem Prozess oder den Ablauf einer Zeit zu warten.

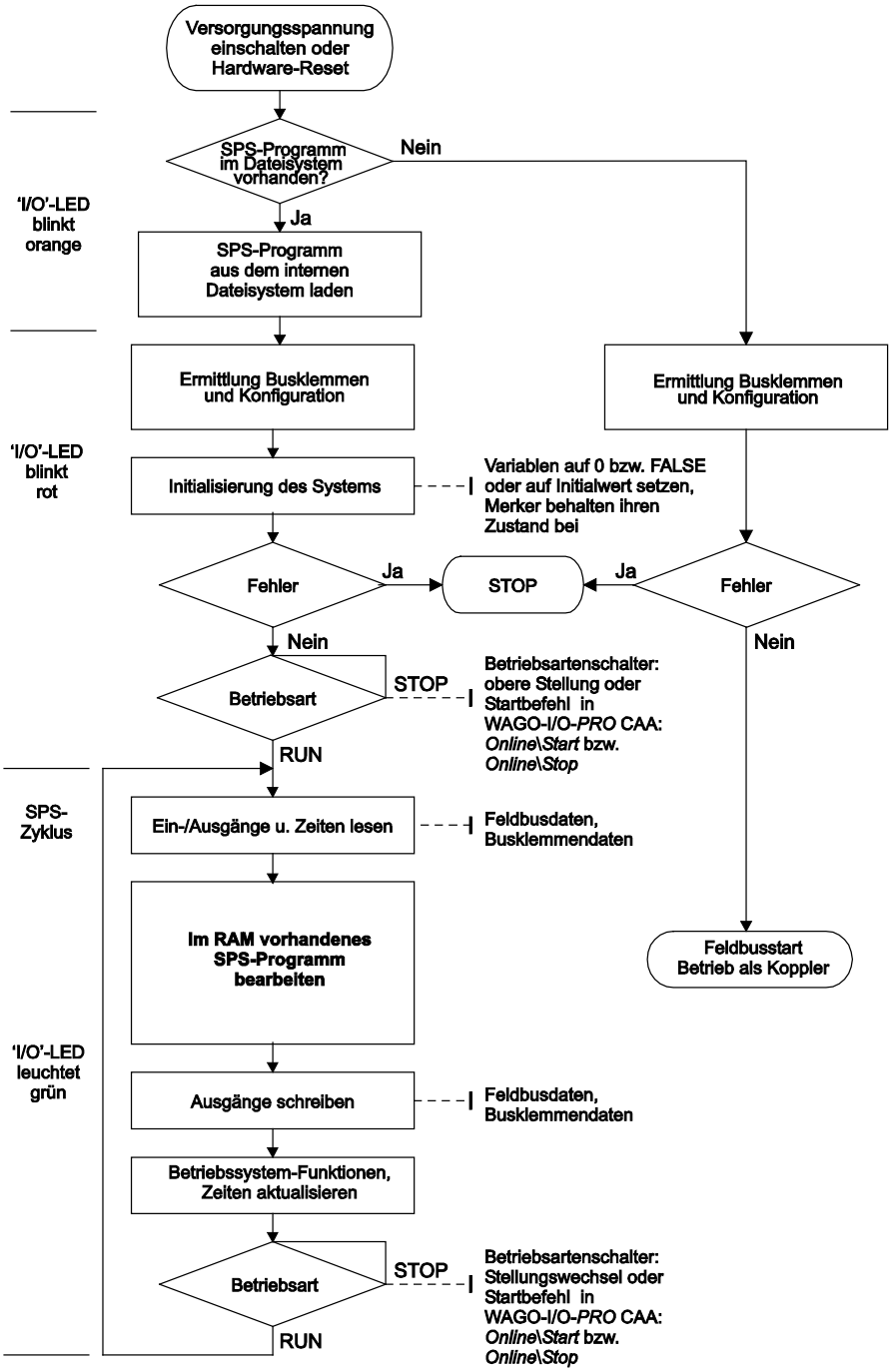


Abbildung 36: Anlauf des Feldbuscontrollers

7.2 Prozessdatenaufbau

7.2.1 Prinzipieller Aufbau

Nach dem Einschalten erkennt der Feldbuscontroller alle im Knoten gesteckten Busklemmen, die Daten liefern bzw. erwarten (Datenbreite/Bitbreite > 0).
In einem Knoten können Analogeingangsklemmen/-ausgangsklemmen und Digitaleingangsklemmen/-ausgangsklemmen sowie Sonderklemmen gemischt angeordnet sein.

Hinweis

Mit Klemmenbusverlängerung bis zu 250 Busklemmen anschließbar!
Mit dem Einsatz der WAGO-Klemmenbusverlängerungs-Kopplerklemme 750-628 und -Endklemme 750-627 ist es möglich, an dem Feldbuscontroller bis zu 250 Busklemmen zu betreiben.

Information

Weitere Information
Die Anzahl der Ein- und Ausgangsbits bzw. -bytes der einzeln angeschalteten Busklemmen entnehmen Sie den entsprechenden Beschreibungen der Busklemmen.

Aus der Datenbreite und dem Typ der Busklemme sowie der Position der Busklemmen im Knoten erstellt der Feldbuscontroller ein internes lokales Prozessabbild. Es ist in einen Eingangs- und Ausgangsdatenbereich unterteilt.

Die Daten der digitalen Busklemmen sind bitorientiert, d. h. der Datenaustausch erfolgt bitweise. Die analogen Busklemmen stehen stellvertretend für alle byteorientierten Busklemmen, bei denen der Datenaustausch also byteweise erfolgt.

Zu diesen Busklemmen gehören z. B. die Zählerklemmen, Busklemmen für Winkel- und Wegmessung sowie die Kommunikationsklemmen.

Für das lokale Ein- und Ausgangsprozessabbild werden die Daten der Busklemmen abhängig von der Reihenfolge ihrer Position am Feldbuscontroller in dem jeweiligen Prozessabbild abgelegt.

Hinweis

Hardware-Änderung kann Änderung des Prozessabbildes bewirken!
Wenn die Hardware-Konfiguration durch Hinzufügen, Austausch oder Entfernen von Busklemmen mit einer Datenbreite > 0 Bit geändert wird, ergibt sich daraus ein neuer Aufbau des Prozessabbildes. Damit ändern sich auch die Adressen der Prozessdaten. Bei einer Erweiterung sind die Prozessdaten aller vorherigen Busklemmen zu berücksichtigen.

Für das Prozessabbild der physikalischen Ein- und Ausgangsdaten steht in dem Controller zunächst jeweils ein Speicherbereich von 256 Worten (Wort 0...255) zur Verfügung.

Für die Abbildung der MODBUS/PFC-Variablen ist der Speicherbereich von jeweils Wort 256...511 reserviert, so dass die MODBUS/PFC-Variablen hinter dem Prozessabbild der Busklemmendaten abgebildet werden.

Ist die Anzahl der Klemmendaten größer als 256 Worte, werden alle darüber hinausreichenden physikalischen Ein- und Ausgangsdaten in einem Speicherbereich an das Ende des bisherigen Prozessabbildes und somit hinten an die MODBUS/PFC-Variablen angehängt (Wort 512...1275).

Im Anschluss an die restlichen physikalischen Busklemmendaten werden die EtherNet/IP PFC-Variablen abgebildet. Dieser Speicherbereich umfasst Wort 1276...1531.

Für zukünftige Protokoll-Erweiterungen ist der anschließende Bereich ab Wort 1532 für die weiteren PFC-Variablen reserviert.

Bei allen WAGO-Feldbuscontrollern ist der Zugriff der SPS auf die Prozessdaten unabhängig von dem Feldbussystem. Dieser Zugriff erfolgt stets über ein anwendungsbezogenes IEC-61131-3-Programm.
Der Zugriff von der Feldbusseite aus ist dagegen feldbusspezifisch.

Für den Feldbuscontroller kann ein MODBUS/TCP-Master über implementierte MODBUS-Funktionen auf die Daten zugreifen, wobei dezimale, bzw. hexadezimale MODBUS-Adressen verwendet werden.
Wahlweise kann der Datenzugriff auch über Ethernet/IP mittels eines Objektmodells erfolgen.

Information

Weitere Information
Eine detaillierte Beschreibung zu diesen feldbusspezifischen Datenzugriffen finden Sie in dem Kapitel „MODBUS-Funktionen“ bzw. in dem Kapitel „EtherNet/IP (Ethernet/Industrial Protocol)“.

Information

Weitere Information
Das feldbusspezifische Prozessabbild ist in den Kapiteln „Busklemmen“ > … > „Aufbau der Prozessdaten…“ für jede WAGO-Busklemme im Einzelnen dargestellt.

7.2.2 Beispiel für ein Eingangsprozessabbild

Im folgenden Bild wird ein Beispiel für ein Prozessabbild mit Eingangsklemmendaten dargestellt.
Die Konfiguration besteht aus 16 digitalen und 8 analogen Eingängen.
Das Eingangsprozessabbild hat damit eine Datenlänge von 8 Worten für die analogen Busklemmen und 1 Wort für die digitalen, also insgesamt 9 Worte.

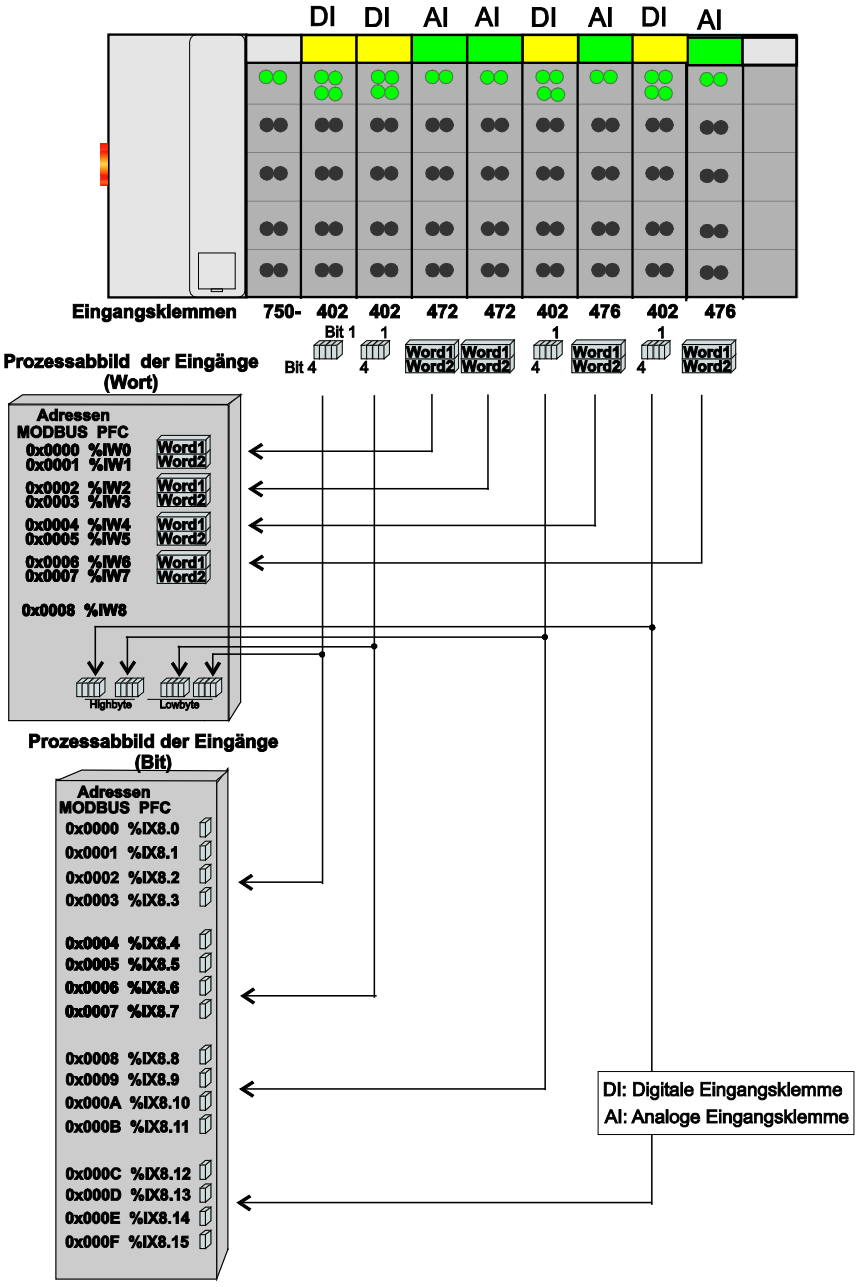


Abbildung 37: Beispiel Eingangsprozessabbild

7.2.3 Beispiel für ein Ausgangsprozessabbild

Als Beispiel für das Prozessabbild mit Ausgangsklemmendaten besteht die folgende Konfiguration aus 2 digitalen und 4 analogen Ausgängen.
Das Ausgangsdaten Prozessabbild besteht aus 4 Worten für die analogen und einem Wort für die digitalen Ausgänge, also insgesamt aus 5 Worten.

Zusätzlich können die Ausgangsdaten mit einem auf die MODBUS-Adresse aufaddierten Offset von 200_{hex} (0x0200) zurückgelesen werden.

Hinweis

Daten > 256 Worte sind mittels aufaddiertem Offset rücklesbar!
Alle Ausgangsdaten, die über 256 Worte hinausreichen und sich deshalb im Speicherbereich 6000_{hex} (0x6000) bis $66F9_{hex}$ (0x66F9) befinden, können mit einem auf die MODBUS-Adresse aufaddierten Offset von 1000_{hex} (0x1000) zurückgelesen werden.

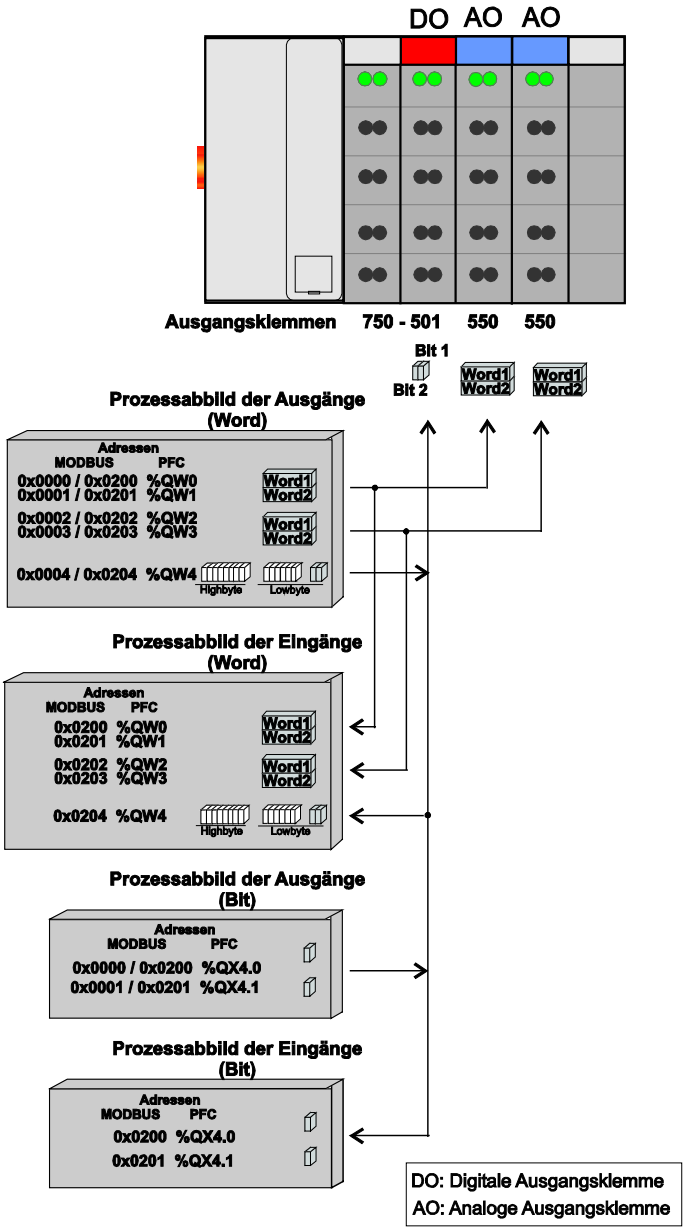


Abbildung 38: Beispiel Ausgangsprozessabbild

7.2.4 Prozessdaten MODBUS/TCP und EtherNet/IP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei dem Feldbuscontroller mit MODBUS und EtherNet/IP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Information

Weitere Information zu dem feldbusspezifischen Prozessdatenaufbau
Der entsprechende feldbusspezifische Aufbau der Prozesswerte aller Busklemmen des WAGO-I/O-SYSTEMs 750 und 753 finden Sie in den Kapiteln „Busklemmen“ > … > „Aufbau der Prozessdaten für MODBUS/TCP“ bzw. „Aufbau der Prozessdaten für EtherNet/IP“.

7.3 Datenaustausch

Der Austausch der Prozessdaten findet bei dem Feldbuscontroller entweder über das MODBUS/TCP-Protokoll bzw. MODBUS/UDP-Protokoll oder über EtherNet/IP statt.

MODBUS/TCP arbeitet nach dem Master-/Slave-Prinzip. Der Master ist eine übergeordnete Steuerung, z. B. ein PC oder eine Speicherprogrammierbare Steuerung.

Die Feldbuscontroller des WAGO-I/O-SYSTEMs 750 sind in der Regel Slave-Geräte. Durch die Programmierung mit IEC 61131-3 können aber auch Controller zusätzlich die Master-Funktion übernehmen.

Der Master fordert die Kommunikation an. Diese Anforderung kann durch die Adressierung an einen bestimmten Knoten gerichtet sein. Die Knoten empfangen die Anforderung und senden, abhängig von der Art der Anforderung, eine Antwort an den Master.

Ein Feldbuscontroller kann eine bestimmte Anzahl gleichzeitiger Verbindungen (Socket-Verbindungen) zu anderen Netzwerkteilnehmern herstellen:

- 3 Verbindung für HTTP (HTML-Seiten von dem Controller lesen),
- 15 Verbindungen über MODBUS/TCP (Ein- und Ausgangsdaten vom Controller lesen oder schreiben),
- 128 Ethernet IP Verbindungen,
- 5 Verbindungen (bei Verwendung der Ethernet.lib) über den PFC (verfügbar in der SPS-Funktionalität für IEC-61131-3-Applikationsprogramme) und
- 2 Verbindungen für WAGO-I/O-*PRO*
  (Diese Verbindungen sind reserviert für das Debuggen des Applikationsprogramms über ETHERNET. WAGO-I/O-*PRO* benötigt für das Debuggen 2 Verbindungen zur selben Zeit. Es kann jedoch nur ein Programmiertool Zugriff auf den Controller haben.)
- 10 Verbindungen für FTP

Die maximale Anzahl der gleichzeitigen Verbindungen kann nicht überschritten werden. Sollen weitere Verbindungen aufgebaut werden, müssen bestehende Verbindungen vorher beendet werden.

Für den Austausch von Daten besitzt der Feldbuscontroller im Wesentlichen drei Schnittstellen:

- die Schnittstelle zum Feldbus (Feldbusmaster)
- die SPS-Funktionalität des Feldbuscontrollers (CPU)
- die Schnittstelle zu den Busklemmen

Ein Datenaustausch findet statt zwischen:

- dem Feldbusmaster und den Busklemmen,
- der SPS-Funktionalität des Feldbuscontrollers (CPU) und den Busklemmen
- dem Feldbusmaster und der SPS-Funktionalität des PFCs (CPU)

Wird der Feldbus MODBUS genutzt, greift der MODBUS-Master über die in dem Controller implementierten MODBUS-Funktionen auf Daten zu, Ethernet/IP hingegen verwendet für den Datenzugriff ein Objektmodell.

Der Zugriff des Feldbuscontrollers auf die Daten erfolgt mit Hilfe eines IEC-61131-3-Applikationsprogramms. Die Adressierung der Daten ist dabei jeweils sehr unterschiedlich.

7.3.1 Speicherbereiche

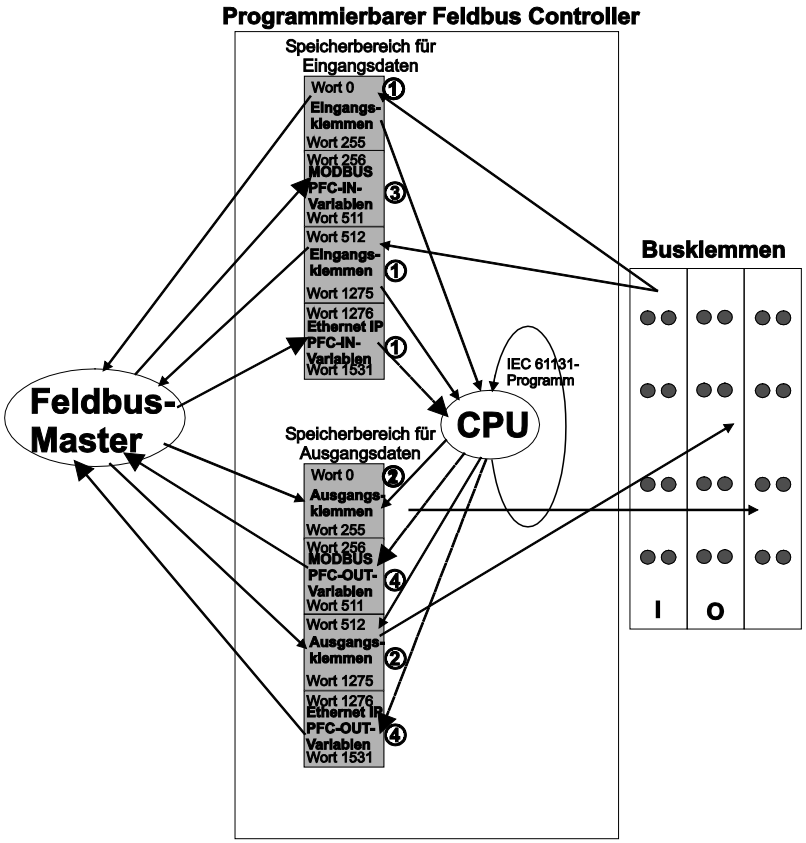


Abbildung 39: Speicherbereiche und Datenaustausch

Das Prozessabbild des Feldbuscontrollers beinhaltet die physikalischen Daten der Busklemmen. Diese belegen im Speicherbereich Wort 0...255 und Wort 512...1275.

① Von der CPU und von der Feldbusseite können die Eingangsklemmendaten gelesen werden.

② Ebenso kann von Seite der CPU und Feldbusseite auf die Ausgangsklemmen geschrieben werden.

In dem jeweils dazwischen liegenden Speicherbereich Wort 256...511 sind die MODBUS-PFC-Variablen abgelegt.

③ Von der Feldbusseite werden die MODBUS-PFC-Eingangsvariablen in den Eingangsspeicherbereich geschrieben und von der CPU zur Verarbeitung eingelesen.

④ Die von der CPU über das IEC-61131-3-Programm verarbeiteten Variablen werden in den Ausgangsspeicherbereich gelegt und können von dem Master ausgelesen werden.

Im Anschluss an die physikalischen Busklemmendaten befindet sich der Speicherbereich Wort 1276...1531 für die EtherNet/IP-PFC-Variablen.

Für zukünftige Protokoll-Erweiterungen und weitere PFC-Variablen ist der anschließende Speicherbereich ab Wort 1532 vorgesehen.

Zusätzlich sind alle Ausgangsdaten auf einen Speicherbereich mit dem Adressen-Offset 0x0200 bzw. 0x1000 gespiegelt. Dadurch ist es möglich, durch Addieren von 0x0200 bzw. 0x1000 zu der MODBUS-Adresse Ausgangswerte zurückzulesen.

In dem Controller sind darüber hinaus weitere Speicherbereiche vorhanden, auf die teilweise von der Feldbusseite aus jedoch nicht zugegriffen werden kann:

- **Datenspeicher (256 kByte)**
  Der Datenspeicher ist ein flüchtiger RAM-Speicher und dient zum Anlegen von Variablen, die nicht zur Kommunikation mit den Schnittstellen sondern für interne Verarbeitungen, wie z. B. die Berechnung von Ergebnissen benötigt werden.

- **Programmspeicher (512 kByte)**
  In dem Programmspeicher wird das IEC-61131-3-Programm abgelegt. Der Code-Speicher ist ein Flash-ROM. Nach dem Einschalten der Versorgungsspannung wird das Programm von dem Flash- in den RAM-Speicher übertragen. Nach fehlerfreiem Hochlauf startet der PFC-Zyklus bei oberer Stellung des Betriebsartenschalters oder durch einen Startbefehl aus WAGO-I/O-PRO.

- **NOVRAM Remanentspeicher (8 kByte)**
  Der Remanentspeicher ist ein nicht flüchtiger Speicher, d. h. nach einem Spannungsausfall bleiben alle Werte der Merker und Variablen beibehalten, die explizit mit „VAR RETAIN“ definiert werden. Die Speicherverwaltung erfolgt automatisch. Der 8 kByte große Speicherbereich teilt sich standardmäßig in einen 4 kByte großen adressierbaren Bereich für die Merker (%MW0 … %MW2045) und einen 4 kByte großen Retain-Bereich für Variablen ohne Speicherbereichsadressierung, die mit „VAR RETAIN“ definiert werden.

Hinweis

Merker nur unter „VAR RETAIN“ remanent!
Beachten Sie, dass die Merker nur remanent sind, wenn Sie diese unter „VAR RETAIN“ deklarieren.

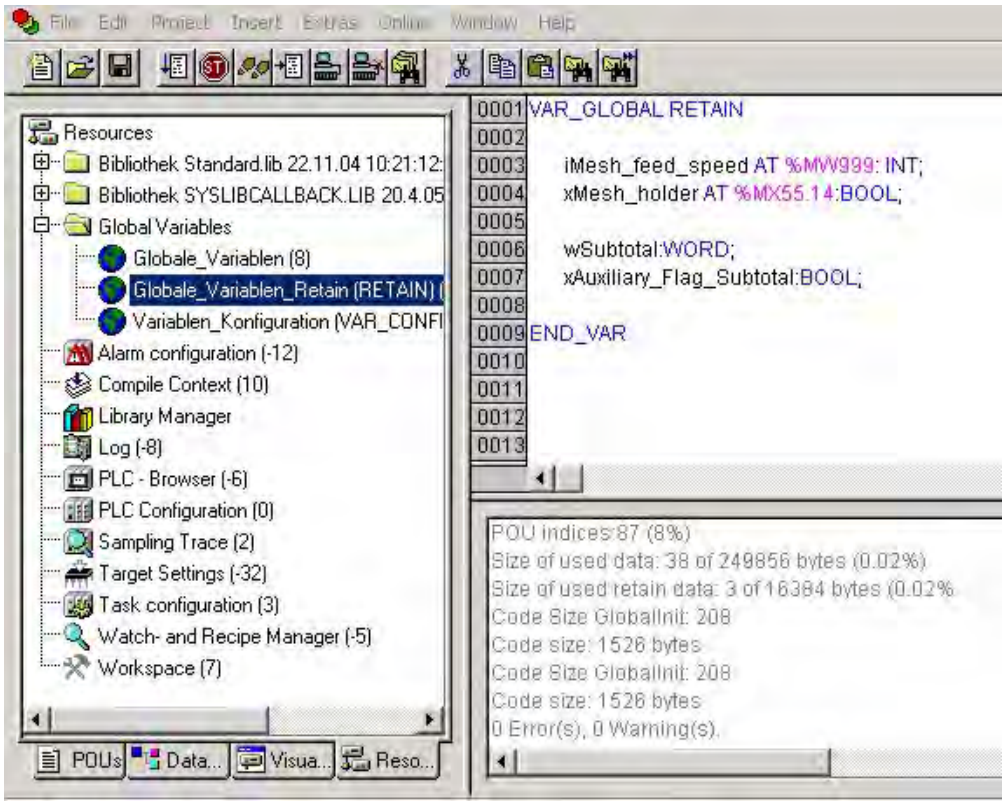


Abbildung 40: Beispiel Deklarierung für remanente Merker unter „VAR RETAIN“

Die Aufteilung des NOVRAM-Remanentspeichers ist variabel (siehe nachfolgenden Hinweis).

Hinweis

NOVRAM-Speicheraufteilung in WAGO-I/O-*PRO* änderbar!
Die Aufteilung des NOVRAM ist in der Programmiersoftware WAGO-I/O-*PRO*/Register „Ressourcen“/Dialogfenster „Zielsystem Einstellungen“ bei Bedarf veränderbar.
Die Startadresse für den Merker-Bereich ist dabei fest adressiert. Die Bereichsgrößen und die Startadresse des Retain-Speichers sind variabel.
Um eine Überlappung der Bereiche auszuschließen, wird jedoch empfohlen, die Standardeinstellung beizubehalten.
Hierbei ist die Größe des Merker-Bereichs mit 16#1000 vorgegeben und daran im Anschluss der Retain-Speicher mit der Größe 16#1000.

7.3.2 Adressierung

Ein- und Ausgänge der Busklemmen an einem Feldbuscontroller werden intern adressiert, sobald sie in Betrieb genommen werden. Die Reihenfolge, in welcher die gesteckten Busklemmen adressiert werden, hängt von der Art der Busklemme (Eingangsklemme, Ausgangsklemme etc.) ab. Aus diesen Adressen baut sich das Prozessabbild zusammen.

Hinweis

Verschiedene Möglichkeiten zur Adressierung der Busklemmen nutzen!
In diesem Kapitel wird die Adressierung und interne Funktionsweise eines Feldbuscontrollers mit gesteckten Busklemmen näher erläutert.
Ein Verständnis der Zusammenhänge ist wichtig, wenn Sie Adressen konventionell über Auszählen zuweisen möchten.
Neben dieser Möglichkeit der Adressierung steht Ihnen der **WAGO-I/O-Konfigurator** zur Verfügung. Dieser unterstützt Sie bei der Adressierung und Protokollzuweisung der gesteckten Busklemmen.
Sie wählen im I/O-Konfigurator die gesteckten Busklemmen aus.
Die korrekte Adressierung übernimmt die Software für Sie.
Der I/O-Konfigurator wird aus der WAGO-I/O-*PRO* heraus gestartet.
Eine nähere Beschreibung lesen Sie in Kapitel „PFC mit WAGO-I/O-*PRO* programmieren“ > … > „Feldbuscontroller mit dem I/O-Konfigurator konfigurieren“.

7.3.2.1 Adressierung der Busklemmen

Bei der Adressierung werden zunächst die komplexen Busklemmen (Busklemmen, die ein oder mehrere Byte belegen) entsprechend ihrer physikalischen Reihenfolge hinter dem Feldbuskoppler/-controller berücksichtigt. Diese belegen somit die Adressen ab Wort 0.

Im Anschluss daran folgen, immer in Bytes zusammengefasst, die Daten der übrigen Busklemmen (Busklemmen, die weniger als ein Byte belegen). Dabei wird entsprechend der physikalischen Reihenfolge Byte für Byte mit diesen Daten aufgefüllt. Sobald ein ganzes Byte durch die bitorientierten Busklemmen belegt ist, wird automatisch das nächste Byte begonnen.

Hinweis

Hardware-Änderung kann Änderung des Prozessabbildes bewirken!
Wenn die Hardware-Konfiguration geändert bzw. erweitert wird, kann sich daraus ein neuer Aufbau des Prozessabbildes ergeben. Damit ändern sich dann auch die Adressen der Prozessdaten. Bei einer Erweiterung sind die Prozessdaten aller vorherigen Busklemmen zu berücksichtigen.

Hinweis

Prozessdatenanzahl beachten!
Entnehmen Sie die Anzahl der Ein- und Ausgangsbits bzw. -bytes für die einzelnen angeschalteten Busklemmen den entsprechenden Beschreibungen der Busklemmen.

Tabelle 29: Datenbreite der Busklemmen (Beispiele)

Datenbreite ≥ 1 Byte/Kanal	Datenbreite = 1 Bit/Kanal
Analogeingangsklemmen	Digitaleingangsklemmen
Analogausgangsklemmen	Digitalausgangsklemmen
Eingangsklemmen für Thermoelemente	Digitalausgangsklemmen mit Diagnose (2 Bit/Kanal)
Eingangsklemmen für Widerstandssensoren	Einspeiseklemmen mit Sicherungshalter/Diagnose
Pulsweitenausgangsklemmen	Solid State Lastrelais
Schnittstellenklemmen	Relais-Ausgangsklemmen
Vor-/Rückwärtszähler	
Busklemmen für Winkel- und Wegmessung	

7.3.2.2 IEC-61131-3-Adressräume

Tabelle 30: IEC-61131-3-Adressräume

Adressraum	MODBUS-Zugriff	SPS-Zugriff	Beschreibung
phys. Eingänge	read	read	Physikalische Eingänge (%IW0...%IW255 und %IW512...%IW1275)
phys. Ausgänge	read/write	read/write	Physikalische Ausgänge (%QW0...%QW255 und %QW512...%QW1275)
MODBUS/TCP PFC-IN-Variablen	read/write	read	Flüchtige SPS-Eingangsvariablen (%IW256...%IW511)
MODBUS/TCP PFC-OUT-Variablen	read	read/write	Flüchtige SPS-Ausgangsvariablen (%QW256...%QW511)
EtherNet/IP-PFC-IN-Variablen	-	read	Flüchtige SPS-Eingangsvariablen (%IW1276 ... %IW1531)
EtherNet/IP-PFC-OUT-Variablen	-	read/write	Flüchtige SPS-Ausgangsvariablen (%QW1276 ... %QW1531)
Konfigurationsregister	read/write	-	siehe Kapitel „MODBUS-Funktionen →MODBUS-Register → Konfigurationsregister“
Firmware-Register	read	-	siehe Kapitel „MODBUS-Funktionen → MODBUS-Register → Firmware-Informationsregister“
Retain-Variablen	read/write	read/write	Remanent-Speicher (%MW0...%MW2045)

7.3.2.3 Absolute Adressierung

Die direkte Darstellung einzelner Speicherzellen (absolute Adressen) nach IEC 61131-3 erfolgt mittels spezieller Zeichenketten:

Tabelle 31: Absolute Adressen

Position	Zeichen	Benennung	Kommentar
1	%	Leitet absolute Adresse ein	
2	I Q M	Eingang Ausgang Merker	
3	X* B W D	Einzelbit Byte (8 Bits) Word (16 Bits) Doubleword (32 Bits)	Datenbreite
4		Adresse	

z. B. wortweise: %QW27 (28. Wort), bitweise: %IX1.9 (10.Bit im 2. Wort)

* Das Kennzeichen ‘X’ für Bits kann entfallen

Hinweis

Zeichenketten ohne Leer- und Sonderzeichen eingeben!
Die Zeichenketten der absoluten Adressen sind zusammenhängend, d. h. ohne Leerzeichen oder Sonderzeichen einzugeben!

Beispieladressierungen:

Tabelle 32: Beispieladressierung

	Eingänge			
Bit	%IX14.0 ... 15		%IX15.0 ... 15	
Byte	%IB28	%IB29	%IB30	%IB31
Wort	**%IW14**		**%IW15**	
Doppelwort	%ID7			

	Ausgänge			
Bit	%QX5.0 ... 15		%QX6.0 ... 15	
Byte	%QB10	%QB11	%QB12	%QB13
Wort	**%QW5**		**%QW6**	
Doppelwort	%QD2 (oberer Teil)		%QD3 (unterer Teil)	

	Merker			
Bit	%MX11.0 ... 15		%MX12.0 ... 15	
Byte	%MB22	%MB23	%MB24	%MB25
Wort	**%MW11**		**%MW12**	
Doppelwort	%MD5 (oberer Teil)		%MD6 (unterer Teil)	

Adressen berechnen (in Abhängigkeit von der Wortadresse):

Bit-Adresse:	Wortadresse .0 bis .15
Byte-Adresse:	1. Byte: 2 x Wortadresse
	2. Byte: 2 x Wortadresse + 1
DWord-Adresse:	Wortadresse (gerade Zahl) / 2
	bzw. Wortadresse (ungerade Zahl) / 2, abgerundet

7.3.3 Datenaustausch MODBUS/TCP-Master und Busklemmen

Der Datenaustausch zwischen MODBUS/TCP-Master und den Busklemmen erfolgt über die in dem Feldbuskoppler/-controller implementierten MODBUS-Funktionen durch bit- oder wortweises Lesen und Schreiben.

Im Feldbuskoppler/-controller gibt es 4 verschiedene Typen von Prozessdaten:

- Eingangsworte
- Ausgangsworte
- Eingangsbits
- Ausgangsbits

Der wortweise Zugriff auf die digitalen Ein- und Ausgangsklemmen erfolgt entsprechend der folgenden Tabelle:

Tabelle 33: Zuordnung digitale Ein-/Ausgänge zum Prozessdatenwort gemäß Intel-Format

Digitale Eingänge/ Ausgänge	16.	15.	14.	13.	12.	11.	10.	9.	8.	7.	6.	5.	4.	3.	2.	1.
Prozessdaten-wort	Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Byte	High-Byte D1								Low-Byte D0							

Durch Addieren eines Offsets von 200 hex (0x0200) zu der MODBUS-Adresse können die Ausgänge zurückgelesen werden.

Hinweis

Daten > 256 Worte sind mittels addiertem Offset rücklesbar!
Alle Ausgangsdaten, die über 256 Worte hinausreichen und deshalb in dem Speicherbereich 0x6000 bis 0x62FC liegen, können mit einem auf die MODBUS-Adresse addierten Offset von 1000_{hex} (0x1000) zurückgelesen werden.

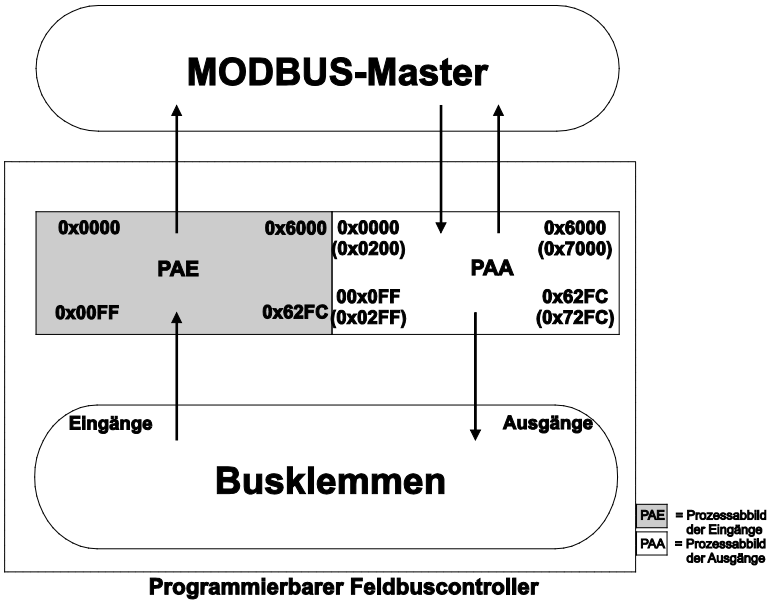


Abbildung 41: Datenaustausch zwischen MODBUS-Master und Busklemmen

Ab Adresse 0x1000 liegen die Registerfunktionen. Diese sind analog mit den implementierten MODBUS-Funktionscodes (read/write) ansprechbar. Anstatt der Adresse eines Klemmenkanals wird dazu die jeweilige Registeradresse angegeben.

Information

Weitere Information

Eine detaillierte Beschreibung der MODBUS-Adressierung ist in dem Kapitel „MODBUS-Register-Mapping“ zu finden.

7.3.4 Datenaustausch EtherNet/IP-Master und Busklemmen

Der Datenaustausch zwischen EtherNet/IP-Master und den Busklemmen ist objektorientiert. Jeder Knoten im Netz wird als Sammlung von Objekten dargestellt.

Das Assembly-Object legt den Aufbau der Objekte für die Datenübertragung fest. Mit dem Assembly-Object können Daten (z. B. I/O-Daten) zu Blöcken zusammengefasst (gemappt) und über eine einzige Nachrichtenverbindung versendet werden. Durch dieses Mapping sind weniger Zugriffe auf das Netzwerk nötig.

Es wird zwischen In- und Output-Assemblies unterschieden.

Eine Input-Assembly liest Daten von der Applikation über das Netz ein bzw. produziert Daten auf dem Netzwerk.

Eine Output-Assembly schreibt Daten an die Applikation bzw. konsumiert Daten vom Netzwerk.

In dem Feldbuskoppler/-controller sind bereits verschiedene Assembly-Instanzen fest vorprogrammiert (statisches Assembly).

Nach Einschalten der Versorgungsspannung werden von dem Assembly-Object Daten aus dem Prozessabbild zusammengefasst. Sobald eine Verbindung aufgebaut ist, kann der Master die Daten mit „Klasse“, „Instanz“ und „Attribut“ adressieren und darauf zugreifen, bzw. mittels I/O-Verbindungen lesen und/oder schreiben.

Das Mapping der Daten ist abhängig von der gewählten Assembly-Instanz des statischen Assembly.

Information

Weitere Information

Die Assembly-Instanzen für das statische Assembly sind in dem Kapitel „EtherNet/IP“ beschrieben.

7.3.5 Datenaustausch SPS-Funktionalität (CPU) und Busklemmen

Die SPS-Funktionalität (CPU) des PFCs hat über absolute Adressen direkten Zugriff auf die Busklemmendaten.

Der PFC spricht die Eingangsdaten mit absoluten Adressen an. Die Daten können dann controller-intern über das IEC-61131-3-Programm verarbeitet werden. Merker werden dabei in einem remanenten Speicherbereich abgelegt. Anschließend können die Verknüpfungsergebnisse direkt über die absolute Adressierung in die Ausgangsdaten geschrieben werden.

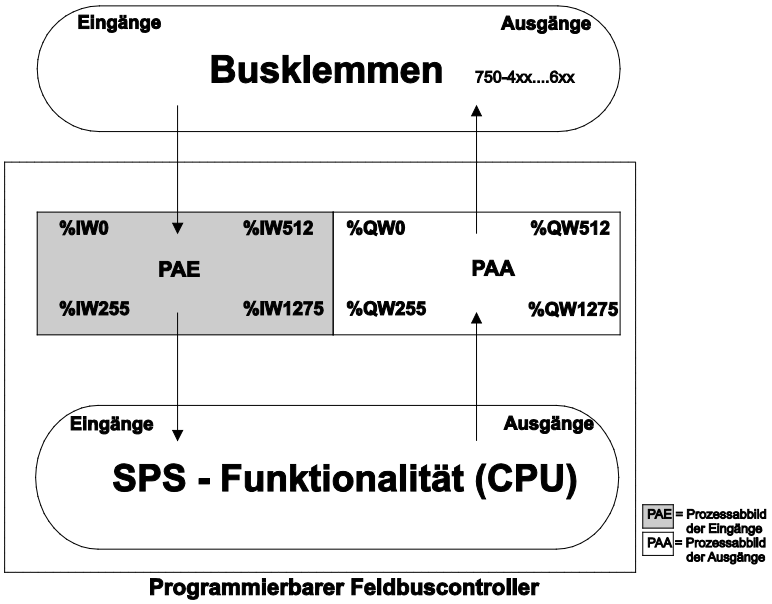


Abbildung 42: Datenaustausch zwischen SPS-Funktionalität (CPU) des PFCs und Busklemmen

7.3.6 Datenaustausch Master und SPS-Funktionalität (CPU)

Der Feldbusmaster und die SPS-Funktionalität (CPU) des Feldbuscontrollers haben unterschiedliche Sichtweisen auf die Daten.

Vom Master erzeugte Variablendaten gelangen als Eingangsvariablen zum Feldbuscontroller und werden dort weiter bearbeitet.
In dem Feldbuscontroller erstellte Daten werden als Ausgangsvariablen über den Feldbus zum Master gesendet.

In dem Feldbuscontroller kann ab Wortadresse 256 bis 511 (Doppelwortadresse 128-255, Byteadresse 512-1023) auf die MODBUS/TCP PFC-Variablendaten zugegriffen werden und ab Wortadresse 1276 bis 1531 (Doppelwortadresse 638-765, Byteadresse 2552-3063) auf die Variablendaten des Feldbuscontrollers.

7.3.6.1 Beispiel MODBUS/TCP-Master und SPS-Funktionalität (CPU)

Datenzugriff vom MODBUS/TCP-Master

Von dem MODBUS-Master wird grundsätzlich wortweise oder bitweise auf die Daten zugegriffen.
Die Adressierung der ersten 256 Datenworte von den Busklemmen beginnt beim wortweisen und bitweisen Zugriff bei 0.
Die Adressierung der Daten von den Variablen beginnt beim wortweisen Zugriff bei Wort 256, beim bitweisen Zugriff erfolgt die Adressierung dann ab:

4096 für Bit 0 im Wort 256
4097 für Bit 1 im Wort 256
...
8191 für Bit 15 im Wort 511.

Die Bit-Nummer lässt sich mit folgender Formel bestimmen:

BitNr = (Wort * 16) + Bitnr_im_Wort
Beispiel: 4097 = (256 * 16) + 1

Datenzugriff von der SPS-Funktionalität (CPU)

Die SPS-Funktionalität des PFCs verwendet bei dem Zugriff auf dieselben Daten eine andere Art der Adressierung. Bei der Deklaration von 16Bit-Variablen ist die SPS-Adressierung identisch mit der wortweisen Adressierung des MODBUS-Masters. Bei der Deklaration von booleschen Variablen (1 Bit) wird im Gegensatz zum MODBUS eine andere Notation verwendet. Hierbei setzt sich die Bitadresse aus den Elementen Wortadresse und Bitnummer im Wort zusammen, getrennt durch einen Punkt.

Beispiel:

Bitzugriff MODBUS auf Bitnummer 4097 => Bitadressierung in der SPS <Wortnr>.<Bitnr> = 256.1

Die SPS-Funktionalität des PFCs kann außerdem byteweise und doppelwortweise auf die Daten zugreifen.

Bei dem byteweisen Zugriff errechnen sich die Adressen nach folgenden Formeln:

High-Byte Adresse = Wortadresse*2
Low-Byte Adresse = (Wortadresse*2) + 1

Bei dem doppelwortweisen Zugriff errechnet sich die Adresse nach folgender Formel:

Doppelwort Adresse = High-Wortadresse/2 (abgerundet)
oder = Low-Wortadresse/2

Information

Weitere Information
Eine detaillierte Beschreibung der MODBUS- und der entsprechenden IEC-61131-Adressierung finden Sie in dem Kapitel „MODBUS-Register-Mapping“.

7.3.7 Anwendungsbeispiel

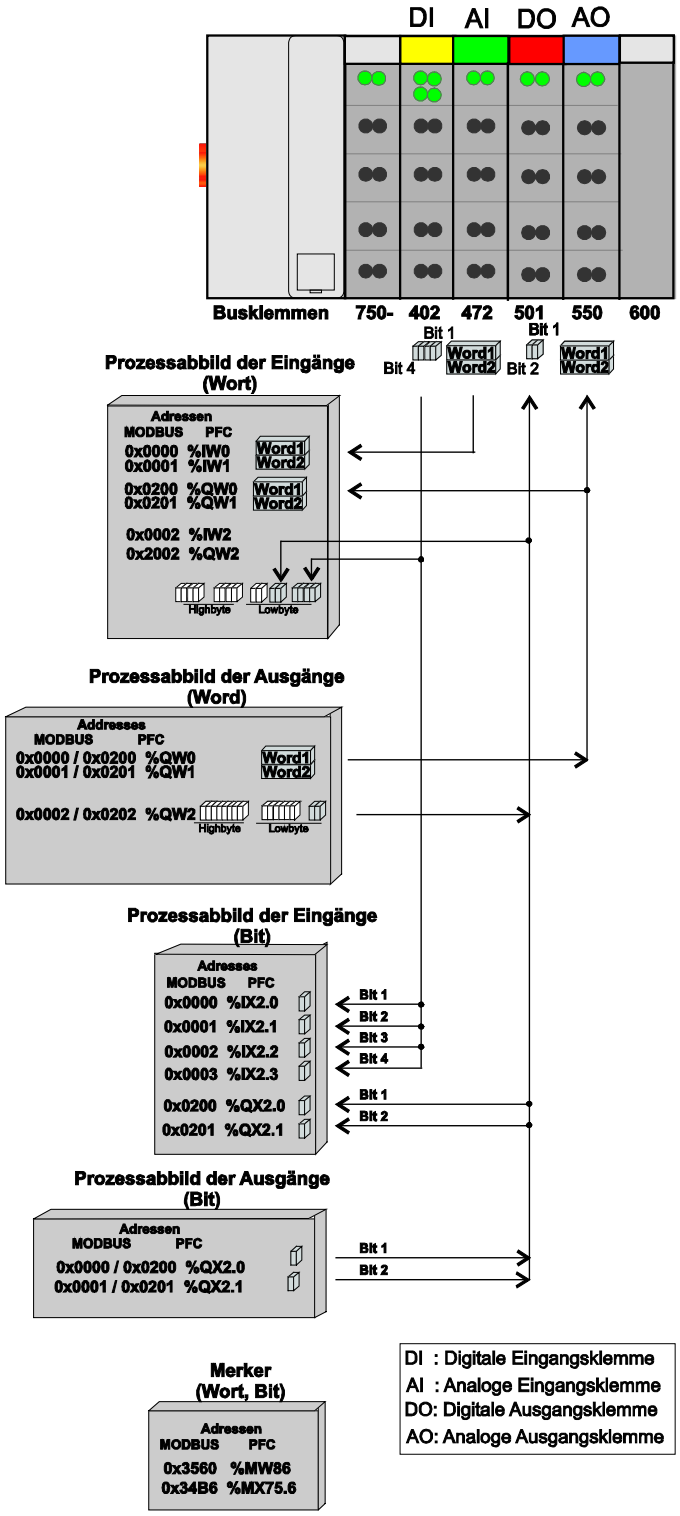


Abbildung 43: Adressierungsbeispiel für einen Feldbusknoten

8 In Betrieb nehmen

In diesem Kapitel wird Ihnen exemplarisch die Vorgehensweise für die Inbetriebnahme eines Feldbusknotens schrittweise aufgezeigt.

Hinweis

Exemplarisches Beispiel!
Diese Beschreibung ist exemplarisch und beschränkt sich hier auf die Ausführung einer lokalen Inbetriebnahme eines einzelnen Feldbusknoten mit einem nicht vernetzten Rechner unter Windows.

Für die Inbetriebnahme sind zwei Arbeitsschritte erforderlich. Die Beschreibung dieser Arbeitsschritte finden Sie in den entsprechenden nachfolgenden Kapiteln.

- **PC und Feldbusknoten anschließen**
- **IP-Adresse an den Feldbusknoten vergeben**

Hinweis

Die IP-Adresse muss im Netzwerk eindeutig sein!
Für eine fehlerfreie Netzwerkkommunikation, beachten Sie, dass die zugewiesene IP-Adresse im Netzwerk eindeutig sein muss!
Im Fehlerfall wird Ihnen beim nächsten Neustart über die I/O-LED die Fehlermeldung „Fehler in der IP-Adresskonfiguration“ (Fehlercode 6 – Fehlerargument 7) angezeigt.

Es gibt verschiedene Möglichkeiten, die IP-Adresse zu vergeben. Diese werden in den nachfolgenden Kapiteln einzeln beschrieben.

Im Anschluss an die Inbetriebnahmekapitel zur Vorbereitung der Kommunikation werden zusätzlich die folgenden Themen beschrieben:

- **Flash-Dateisystem vorbereiten**
- **Echtzeituhr synchronisieren**
- **Werkseinstellungen wiederherstellen**

Nach den oben genannten Themen finden Sie Hinweise zur Programmierung des Feldbuscontrollers mit WAGO-I/O-*PRO* und die Beschreibung der internen Webseiten des Web-based Management-Systems (WBM) für weitere Einstellungen des Feldbuscontrollers.

8.1 PC und Feldbusknoten anschließen

1. Montieren Sie den Feldbusknoten auf der Hutschiene.
   Beachten Sie hierbei die Montagehinweise gemäß dem Kapitel „Montieren“.

2. Schließen Sie die 24V-Versorgungsspannung an die Einspeiseklemmen an.

3. Verbinden Sie eine ETHERNET-Schnittstelle des PCs mit einer ETHERNET-Schnittstelle des Feldbuscontrollers

4. Schalten Sie die Betriebsspannung ein.
   Achten Sie dabei darauf, dass der Betriebsartenschalter nicht in der unteren Position steht.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot. Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, wird dieser mittels der I/O-LED durch rotes Blinken als Fehlercode ausgegeben.
Wird nach Anlauf des Feldbuskopplers über die I/O-LED durch 6-maliges rotes Blinken der Fehlercode 6 und anschließend durch 4-maliges rotes Blinken das Fehlerargument 4 ausgegeben, zeigt dieses an, dass noch keine IP-Adresse zugewiesen wurde.

8.2 IP-Adresse an den Feldbusknoten vergeben

- **IP-Adresse mit dem Adresswahlschalter vergeben**
  (manuell über den DIP-Schalter)
- **IP-Adresse mit DHCP vergeben**
  (IP-Adresse über den Feldbus)
- **IP-Adresse mit WAGO-ETHERNET-Settings vergeben**
  (statische IP-Adresse über die serielle Kommunikationsschnittstelle)
- **IP-Adresse über ein SPS-Programm vergeben**
  (statische IP-Adresse)
- **IP-Adresse mit einem BootP-Server vergeben**
  (statische IP-Adresse über den Feldbus)

8.2.1 IP-Adresse mit dem Adresswahlschalter vergeben

Über den Adresswahlschalter können Sie die Host-ID, d. h. das letzte Byte der IP-Adresse, die in dem Web-Based-Management-System auf der Seite „**TCP/IP**“ unter dem Eintrag „**Switch IP-Address**“ eingetragen ist, mit Werten zwischen 1 und 254 binär codiert einstellen.

Beispiel:
Im Feldbuscontroller gespeicherte DIP-Switch-IP-Adresse: 192.168.1
Eingestellter DIP-Schalterwert: **50** (00110010_{bin})
Resultierende IP-Adresse für den Feldbuscontroller: 192.168.1.**50**

Hinweis

Host-ID 1... 254 über den Adresswahlschalter frei einstellbar!
Stellen Sie über den Adresswahlschalter das letzte Byte der IP-Adresse auf einen Wert zwischen 1 und 254, dann ist der DIP-Schalter aktiv, und die IP-Adresse setzt sich aus der im Feldbuscontroller gespeicherten DIP-Switch-Basisadresse und der am DIP-Schalter eingestellten Host-ID zusammen.
Die IP-Adresse, welche z. B. über das Web-Based-Management-System oder WAGO-ETHERNET-Settings eingestellt wurde, ist inaktiv.

Hinweis

Adresswahlschalterwerte 0 und 255 sind fest vordefiniert, Adresswahlschalter inaktiv!
Stellen Sie über den Adresswahlschalter den Wert 0 oder 255 ein, so ist der Adresswahlschalter inaktiv und es wird die im Feldbuscontroller konfigurierte Einstellung verwendet.
Bei dem Wert 0 gelten dann die Einstellungen des Web-Based-Management-Systems oder ETHERNET-Settings.
Stellen Sie den Wert 255 ein, so ist die Konfiguration über DHCP aktiviert.

Die verwendete Basisadresse besteht aus den ersten drei Bytes der IP-Adresse. Diese hängt immer von der aktuell im Feldbuscontroller gespeicherten DIP-Switch-IP-Adresse ab.
Die Einstellung des Adresswahlschalters auf 1 … 254 überschreibt entsprechend den Wert der Host-ID.

Sollte in dem Feldbuscontroller noch keine statische IP-Adresse vorhanden sein, wird bei der DIP-Schaltereinstellung auf 1 ... 254 der firmwareseitig fest vorgegebene Standardwert **192.168.1.** als Basisadresse verwendet.

Information

Weitere Informationen zum Ändern der statischen Basisadresse
Sie können die aktuell im Feldbuscontroller gespeicherte Basisadresse nach Wunsch auch ändern. Dazu gehen Sie, wie in dem nachfolgenden Kapitel „IP-Adresse mit WAGO-ETHERNET-Settings vergeben“ beschrieben, vor.

1. Um die IP-Adresse über den Adresswahlschalter durch Einstellen der Host-ID (letzte Stelle der IP-Adresse) mit einem Wert ungleich 0/255 einzustellen, rechnen Sie zunächst die Host-ID in die Binärdarstellung um. Beispielsweise ergibt die Host-ID 50 eine binäre Codierung von 00110010.

2. Stellen Sie die Bits der Reihe nach mit den 8 Adressschaltern ein. Beginnen Sie mit Adressschalter 1 zur Einstellung von Bit 0 (LSB) und enden Sie bei Adressschalter 8 mit Bit 7 (MSB).

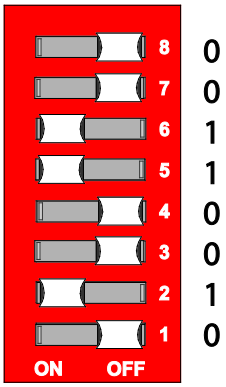


Abbildung 44: Adresswahlschalter, hier z. B. Einstellung des Wertes „50“ ($2^1 + 2^4 + 2^5$)

3. Führen Sie nach dem Verstellen des Adresswahlschalters einen Neustart des Feldbuskopplers durch, damit die geänderte Konfiguration übernommen wird.

8.2.2 IP-Adresse mit DHCP vergeben

Wenn Sie die IP-Adresse mit DHCP zuweisen möchten, erfolgt dieses automatisch über einen im Netz vorhandenen DHCP-Server.

Hinweis

Totaler Netzwerkausfall bei zwei DHCP-Servern in einem Netzwerk!
Damit es nicht zu einem Netzwerkausfall kommt, schließen Sie niemals einen PC, auf dem ein DHCP-Server installiert ist, an ein globales Netzwerk an. In größeren Netzwerken ist in der Regel bereits ein DHCP-Server vorhanden, mit dem es zu Kollisionen kommt, wonach das Netzwerk zusammenbricht.

Hinweis

Für weitere Konfiguration muss ein DHCP-Server im Netz sein!
Installieren Sie in Ihrem lokalen Netzwerk einen DHCP-Server auf Ihren PC, sofern dieser noch nicht vorhanden ist. Sie können einen DHCP-Server kostenlos aus dem Internet herunterladen, z. B. unter: http://windowspedia.de/dhcp-server_download/

Hinweis

DHCP-Server feste IP-Adresse zuweisen und auf gemeinsames Subnetz achten!
Beachten Sie, dass der DHCP-Server eine feste IP-Adresse haben muss, und dass Feldbusknoten und DHCP-Server sich in demselben Subnetz befinden müssen.

Hinweis

Über DHCP-Server bezogene IP-Adressen sind nur temporär gültig!
Beachten Sie, dass eine über DHCP-Server bezogene IP-Adresse zeitlich begrenzt gültig ist. Wenn der DHCP-Server nach Ablauf der Nutzungsdauer nicht verfügbar ist, gibt der Feldbusknoten die IP-Adresse frei und ist danach nicht mehr erreichbar!

Die Beschreibung umfasst die folgenden Arbeitsschritte:

- DHCP aktivieren
- IP-Adresse dauerhaft vergeben durch Option „use IP from EEPROM“

8.2.2.1 DHCP aktivieren

Hinweis

Für aktive Software-Konfiguration, Adresswahlschalter auf 255 stellen!
Stellen Sie den Adresswahlschalter auf 255, damit die Adresswahl per DIP-Schalter deaktiviert und DHCP aktiviert wird.
Führen Sie nach dem Verstellen des Adresswahlschalters einen Neustart des Feldbusknotens durch, damit die geänderte Konfiguration übernommen wird.

Alternativ kann die DHCP-Anfrage auch auf den internen Web-Seiten aktiviert werden oder über WAGO-ETHERNET-Settings.

Hinweis

DHCP auf den Web-Seiten oder über WAGO-ETHERNET-Settings aktivieren!
Falls bereits über eine IP-Adresse ein Zugriff auf die internen Web-Seiten des WBMs möglich ist, aktivieren Sie DHCP auf der HTML-Seite „Port“, damit über DHCP eine neue IP-Adresse zugewiesen wird.
Im Auslieferungszustand ist DHCP standardmäßig nicht aktiviert.
Ansonsten können Sie DHCP auch über WAGO-ETHERNET-Settings, in dem Register **Netzwerk**, aktivieren.

Nach dem Start des Feldbusknotens, erfolgt automatisch die Zuweisung einer IP-Adresse.

Information

Weitere Informationen zum Auslesen der IP-Adresse
Sie können über die Service-Schnittstelle mittels WAGO-ETHERNET-Settings, über die Schaltfläche **[Identifizieren],** die aktuell zugewiesene IP-Adresse auslesen.

8.2.2.2 IP-Adresse dauerhaft durch Option „use IP from EEPROM“

Hinweis

Für die dauerhafte Adressvergabe muss die im EEPROM gespeicherte IP verwendet werden!
Damit die über DHCP erhaltene IP-Adresse dauerhaft in den Feldbuscontroller übernommen wird, müssen Sie die Option „use IP from EEPROM“ auswählen.
Damit wird die IP-Adresse automatisch als statische Adresse eingetragen und verwendet.

Das Umschalten auf die Option „use IP from EEPROM“ nehmen Sie im **Web-Based-Management-System** vor.

1. Öffnen Sie auf Ihrem PC einen **Web-Browser** (z. B. Microsoft Internet Explorer) für die Anzeige der Feldbuscontroller-internen HTML-Seiten (Web-Based- Management-System).

2. Geben Sie die **IP-Adresse** Ihres Feldbusknotens in das Adressfeld des Browsers ein und drücken Sie die Taste **[Enter]**.

3. Wenn ein Dialogfenster mit einer Passwort-Abfrage erscheint, geben Sie als Administrator den Benutzernamen **„admin“** und das Kennwort **„wago“** ein.

   Dieses dient der Zugriffssicherung und enthält die drei verschiedenen Benutzergruppen: „admin“, „guest“ und „user“.

   In dem Browser-Fenster wird eine Startseite mit den Informationen zu

Ihrem Feldbuscontroller angezeigt (Startseite auf Seite „Security“ änderbar). Über Hyperlinks in der linken Navigationsleiste gelangen Sie zu den weiteren Informationen.

4. Klicken Sie in der linken Navigationsleiste auf den Link **[Port]**, um die HTML-Seite für die Protokollauswahl zu öffnen.

Sie erhalten eine Liste aller Protokolle, die der Feldbuscontroller unterstützt.

5. Wählen Sie die Option „use IP from EEPROM“.
   Sie haben damit zugleich die Nutzung des DHCP-Servers deaktiviert.

6. Bestätigen Sie Ihre Änderungen mit der Schaltfläche **[SUBMIT]**.

7. Damit die Einstellungen übernommen werden, führen Sie anschließend einen Neustart durch.

8. Für einen Software-Reset klicken Sie in der linken Navigationsleiste auf den Link **[Security],** um die HTML-Seite zu öffnen, auf der Sie Passwörter einrichten und einen Software-Reset auslösen können.

9. Klicken Sie unten auf der Seite auf die Schaltfläche **[Software Reset].**

   Der Feldbuskoppler/-controller startet anschließend mit den Konfigurationen, die zuvor ins EEPROM geladen wurden, die Verbindung zum Browser wird unterbrochen.

10. Wollen Sie erneut über den Browser auf das Gerät zugreifen, müssen Sie nun die geänderte IP-Adresse verwenden.

8.2.3 IP-Adresse mit WAGO-ETHERNET-Settings vergeben

Dieses Programm dient dazu, über die serielle Kommunikationsschnittstelle eine IP-Adresse zu konfigurieren, die Parameter von Feldbuskopplern-/controllern auf Werkseinstellungen zurückzusetzen sowie das Flash-Dateisystem, auf welchem die HTML-Seiten des Feldbuskopplers-/controllers abgelegt sind, zu löschen und zu entpacken.
Zur Datenkommunikation können WAGO-Kommunikationskabel oder WAGO-Funkadapter verwendet werden.

ACHTUNG

Kommunikationskabel nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 bzw. 750-923 nicht unter Spannung!
Der Feldbuscontroller muss dazu spannungsfrei sein!

Hinweis

Für aktive Software-Konfiguration, Adresswahlschalter auf 0 stellen!
Stellen Sie den Adresswahlschalter auf 0, damit der DIP-Schalter deaktiviert wird und die Software-Konfiguration über ETHERNET-Settings aktiviert wird.
Führen Sie nach dem Verstellen des Schalters einen Neustart des Feldbusknotens durch, damit die geänderte Konfiguration übernommen wird.

1. Verbinden Sie Ihren PC unter Verwendung eines WAGO-Kommunikationskabels 750-920 bzw. 750-923 mit der Service-Schnittstelle des Feldbuscontrollers.
2. Starten Sie das Programm **WAGO-ETHERNET-Settings**.
3. Klicken Sie auf **[Identifizieren]**, um den angeschlossenen Feldbusknoten einzulesen und zu identifizieren.
4. Wählen Sie das Register **Netzwerk**.

5. Damit Sie eine feste Adresse vergeben können, wählen Sie im Feld **Bezugsquelle** “Statische Konfiguration“ aus (standardmäßig ist BootP aktiviert).

6. Geben Sie die gewünschte **IP-Adresse** und gegebenenfalls die Adresse der Subnetzmaske und des Gateways ein.

7. Klicken Sie auf die Schaltfläche **[Schreiben]**, um die Einstellungen in den Feldbusknoten zu übernehmen.

8. Sie können nun WAGO-ETHERNET-Settings schließen oder bei Bedarf direkt im Web-Based-Management-System weitere Einstellungen vornehmen. Klicken Sie dazu auf die Schaltfläche **[WBM]** auf der rechten Seite.

8.2.4 IP-Adresse über ein SPS-Programm vergeben

Über ein SPS-Programm kann eine feste IP-Adresse vergeben werden.

Die Vergabe mittels SPS-Programm wird durch den Funktionsblock „Ethernet_Set_Network_Config“ der Bibliothek „Ethernet.lib“ realisiert, welcher in WAGO-I/O-*PRO* eingebunden wird.

Information

Weitere Informationen zur IP-Adressvergabe über ein SPS-Programm!
Eine detaillierte Beschreibung der Bibliothek für die Adressvergabe über ein SPS-Programm entnehmen Sie dem Handbuch zu der WAGO-I/O-*PRO*-Bibliothek „Ethernet.lib“.
Dieses finden Sie auf der Internetseite http://www.wago.com unter Downloads → Zusätzliche Dokumentation und Information für Automatisierungsprodukte → WAGO Software → WAGO-I/O-PRO/CODESYS→ Weitere Informationen → Bibliotheken → Ethernet.lib.

8.2.5 IP-Adresse mit einem BootP-Server vergeben

Mittels eines BootP-Servers kann eine feste IP-Adresse vergeben werden.

Die Vergabe der IP-Adresse mittels BootP-Server ist abhängig von dem jeweiligen BootP-Programm. Die Handhabung ist dem entsprechenden Handbuch zu diesem Programm oder den entsprechend eingebundenen Hilfetexten zu entnehmen.

Hinweis

Für aktive Software-Konfiguration, Adresswahlschalter auf 0 stellen!
Stellen Sie den Adresswahlschalter auf 0, damit der DIP-Schalter deaktiviert wird. Falls bereits über eine IP-Adresse ein Zugriff auf die internen Web-Seiten des WBMs möglich ist, kann anschließend die Software-Konfiguration über das WBM erfolgen.
Im WBM ist standardmäßig BootP aktiviert.
Falls noch keine IP-Adresse zugewiesen ist, können Sie ansonsten WAGO-ETHERNET-Settings verwenden oder auch BootP über WAGO-ETHERNET-Settings, in dem Register **Netzwerk**, aktivieren.
Führen Sie nach dem Verstellen des Adresswahlschalters einen Neustart des Feldbusknotens durch, damit die geänderte Konfiguration übernommen wird.

Hinweis

IP-Adressvergabe nicht über Router möglich!
Die Vergabe der IP-Adresse erfolgt über ein Straight-Through-Kabel, Switches, Hubs oder in einer direkten Verbindung mittels eines Cross-over-Kabels. Über einen Router ist keine Adressenvergabe möglich.

Hinweis

BootP muss im Web-Based-Management-System aktiviert sein!
Beachten Sie, dass BootP auf den internen Webseiten des WBM, auf der HTML-Seite „Port“, aktiviert sein muss.

Im Auslieferungszustand ist BootP standardmäßig aktiviert.

Information **Weitere Information**
Die IP-Adressvergabe mittels BootP-Server ist unter Windows- und Linux-Betriebssystemen möglich. Neben dem WAGO-BootP-Server können beliebige andere BootP-Server verwendet werden.

Information **Weitere Informationen zu dem WAGO-BootP-Server!**
Sie erhalten den „WAGO-BootP-Server 759-315“ kostenlos auf der Internetseite http://www.wago.com .

Die Beschreibung umfasst die folgenden Arbeitsschritte:

- MAC-ID ermitteln
- IP-Adresse ermitteln
- IP-Adresse vergeben und BootP aktivieren
- IP-Adresse dauerhaft vergeben durch Option „use IP from EEPROM“

8.2.5.1 MAC-ID ermitteln

1. Notieren Sie die MAC-ID Ihres Feldbuscontrollers, bevor Sie Ihren Feldbusknoten aufbauen.
   Ist der Feldbuscontroller bereits verbaut, schalten Sie die Betriebsspannung des Feldbuscontrollers aus und nehmen Sie ihn aus dem Verbund heraus.

Die MAC-ID ist auf der Rückseite des Feldbuscontrollers oder auf einem Papierstreifen mit zwei selbstklebenden Abreißetiketten seitlich auf dem Feldbuscontroller aufgebracht.

MAC-ID des Feldbuscontrollers: 0 0 : 3 0 : D E : _ _ : _ _ : _ _

2. Stecken Sie den Feldbuscontroller in den Verbund des Feldbusknotens.

3. Schließen Sie den Feldbusanschluss Ihres mechanisch und elektrisch montierten Feldbusknotens mit dem Feldbuskabel an eine entsprechende freie Schnittstelle Ihres PCs an.
   Der PC muss für diesen Anschluss über eine Netzwerkkarte verfügen. Die Übertragungsrate ist dann abhängig von der Netzwerkkarte Ihres PCs.

4. Starten Sie den PC, der die Funktion des Masters und BootP-Servers übernimmt.

5. Schalten Sie die Spannungsversorgung am Feldbuscontroller (DC-24V-Netzteil) ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.

Während des Hochlaufens blinkt die I/O-LED rot. Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information

Weitere Informationen zu der LED-Signalisierung

Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

Wird nach Anlauf des Feldbuscontrollers durch 6-maliges rotes Blinken der Fehlercode 6 und anschließend durch 4-maliges rotes Blinken das Fehlerargument 4 mittels I/O-LED ausgegeben, zeigt dies an, dass noch keine IP-Adresse zugewiesen wurde.

8.2.5.2 IP-Adresse ermitteln

1. Ist der PC bereits in ein IP-Netzwerk eingebunden, können Sie die IP-Adresse des PCs ermitteln, indem Sie auf Ihrer Bildschirmoberfläche über das **Startmenü / Einstellungen** gehen und auf **Systemsteuerung** klicken.

2. Klicken Sie doppelt auf das Icon **Netzwerk**.
   Das Netzwerk-Dialogfenster wird geöffnet.

   Unter Windows 2000/XP:

   - Wählen Sie **[Netzwerk- und DFÜ-Verbindungen]**.
   - In dem sich öffnenden Fenster klicken Sie mit der rechten Maustaste auf **[LAN-Verbindung]** und öffnen die **Eigenschaften** der Verbindung.
   - Markieren Sie den Eintrag **Internetprotokoll TCP/IP**.

   Unter Windows 7:

   - Wählen Sie in der Systemsteuerung **[Netzwerk- und Freigabecenter]**.
   - In dem sich öffnenden Fenster klicken Sie mit der rechten Maustaste auf **[LAN-Verbindung]** und öffnen die **Eigenschaften** der Verbindung.
   - Markieren Sie den Eintrag **Internetprotokoll V4**.

Hinweis

TCP/IP-Komponente bei Bedarf nachinstallieren!
Fehlt der Eintrag „Internetprotokoll TCP/IP“, installieren Sie die entsprechende TCP/IP-Komponente, und starten Sie Ihren PC neu.
Für die Installation benötigen Sie die Installations-CD für Windows 2000, XP oder 7.

3. Klicken Sie anschließend auf die Schaltfläche **[Eigenschaften...]**.

4. In dem Eigenschaftenfenster entnehmen Sie die IP-Adresse, die Subnetzmaske und gegebenenfalls die Adresse für das Gateway Ihres PCs, und notieren Sie diese Werte:

   IP-Adresse PC: ___ . ___ . ___ . ___
   Subnetzmaske: ___ . ___ . ___ . ___
   Gateway: ___ . ___ . ___ . ___

5. Wählen Sie nun eine gewünschte IP-Adresse für Ihren Feldbusknoten.

Hinweis

PC feste IP-Adresse zuweisen und auf gemeinsames Subnetz achten!
Beachten Sie, dass der PC, auf dem der BootP-Server ausgeführt wird, eine feste IP-Adresse haben muss, und dass der Feldbusknoten und der PC sich in demselben Subnetz befinden müssen.

6. Notieren Sie sich die von Ihnen gewählte IP-Adresse:

   IP-Adresse Feldbusknoten: _ _ _ . _ _ _ . _ _ _ . _ _ _

8.2.5.3 IP-Adresse vergeben

1. Vergeben Sie die gewünschte IP-Adresse für Ihren Feldbusknoten entsprechend der Handhabung, die abhängig von dem eingesetzten BootP-Programm ist.

2. Aktivieren Sie den Frage-Antwort-Mechanismus des BootP-Protokolls entsprechend der Handhabung, die abhängig von dem eingesetzten BootP-Programm ist. Alternativ nehmen Sie die Aktivierung von BootP in WAGO-ETHERNET-Settings oder falls bereits über eine IP-Adresse ein Zugriff auf die internen Web-Seiten des WBMs möglich ist, im Web-Based-Management-System vor.

3. Damit die neue IP-Adresse übernommen wird, führen Sie einen Neustart Ihres Feldbusknoten durch, z. B. mit einem Hardware-Reset durch Unterbrechen der Spannungsversorgung für ca. 2 Sekunden.

8.2.5.4 IP-Adresse dauerhaft durch Option „use IP from EEPROM“

Bei aktiviertem BootP-Protokoll erwartet der Feldbuscontroller die permanente Anwesenheit eines BootP-Servers. Ist jedoch nach einem Power-On-Reset kein BootP-Server verfügbar, dann bleibt das Netzwerk inaktiv.

Hinweis

Für die dauerhafte Adressvergabe muss die im EEPROM gespeicherte IP verwendet werden!
Damit die über DHCP erhaltene IP-Adresse dauerhaft in den Feldbuscontroller übernommen wird, müssen Sie die Option „use IP from EEPROM“ auswählen.
Damit wird die IP-Adresse automatisch als statische Adresse eingetragen und verwendet.

Das Umschalten auf die Option „use IP from EEPROM“ nehmen Sie im **Web-Based-Management-System** vor.

1. Öffnen Sie auf Ihrem PC einen **Web-Browser** (z. B. Microsoft Internet Explorer) für die Anzeige der Feldbuscontroller-internen HTML-Seiten (Web-Based- Management-System).

2. Geben Sie die **IP-Adresse** Ihres Feldbusknotens in das Adressfeld des Browsers ein und drücken Sie die Taste **[Enter]**.

3. Wenn ein Dialogfenster mit einer Passwort-Abfrage erscheint, geben Sie als Administrator den Benutzernamen **„admin“** und das Kennwort **„wago“** ein.

   Dieses dient der Zugriffssicherung und enthält die drei verschiedenen Benutzergruppen: „admin“, „guest“ und „user“.

   In dem Browser-Fenster wird eine Startseite mit den Informationen zu Ihrem Feldbuscontroller angezeigt (Startseite auf Seite „Security“ änderbar). Über Hyperlinks in der linken Navigationsleiste gelangen Sie zu den weiteren Informationen.

4. Klicken Sie in der linken Navigationsleiste auf den Link **[Port]**, um die HTML-Seite für die Protokollauswahl zu öffnen.

Sie erhalten eine Liste aller Protokolle, die der Feldbuscontroller unterstützt.

5. Wählen Sie die Option „use IP from EEPROM“.
   Sie haben damit zugleich die BootP-Abfrage deaktiviert.

6. Bestätigen Sie Ihre Änderungen mit der Schaltfläche **[SUBMIT]**.

7. Damit die Einstellungen übernommen werden, führen Sie anschließend einen Neustart durch.

8. Für einen Software-Reset klicken Sie in der linken Navigationsleiste auf den Link **[Security],** um die HTML-Seite zu öffnen, auf der Sie Passwörter einrichten und einen Software-Reset auslösen können.

9. Klicken Sie unten auf der Seite auf die Schaltfläche **[Software Reset].**

   Der Feldbuskoppler/-controller startet anschließend mit den Konfigurationen, die zuvor ins EEPROM geladen wurden, die Verbindung zum Browser wird unterbrochen.

10. Wollen Sie erneut über den Browser auf das Gerät zugreifen, müssen Sie nun die geänderte IP-Adresse verwenden.

8.2.5.5 Gründe für eine fehlgeschlagene IP-Adressvergabe

- Der PC, auf dem der BootP-Server läuft, befindet sich nicht im selben Subnetz wie der Feldbuscontroller, d. h. die IP-Adressen passen nicht zusammen.

  Beispiel:
  Subnetz-Maske: 255.255.255.0 (Defaultwert eines Feldbuscontrollers)
  PC-IP: 192.168.**2**.100
  Feldbuscontroller-IP: 192.168.**1**.200
  Aufgrund der Subnetzmaske müssen die ersten 3 Stellen der IP-Adressen übereinstimmen.

- PC und/oder Feldbuscontroller haben keine ETHERNET-Verbindung

- Die Signalqualität ist schlecht (Switches oder Hubs verwenden)

8.3 Funktion des Feldbusknotens testen

Information

Weitere Informationen zum Auslesen der IP-Adresse
Sie können über die Service-Schnittstelle mittels WAGO-ETHERNET-Settings, über die Schaltfläche **[Identifizieren],** die aktuell zugewiesene IP-Adresse auslesen.

1. Um die korrekte Vergabe der IP-Adresse und die Kommunikation mit dem Feldbusknoten zu testen, schalten Sie zunächst die Betriebsspannung des Feldbusknotens aus.

2. Stellen Sie eine nicht-serielle Feldbusverbindung zwischen Client-PC und Feldbusknoten her.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot. Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information

Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

3. Rufen Sie die DOS-Eingabeaufforderung unter **Startmenü / Programme / Eingabeaufforderung** auf.

4. Geben Sie den Befehl **ping** mit der von Ihnen vergebenen IP-Adresse in der folgenden Schreibweise ein:

   ping [Leerzeichen] XXX . XXX . XXX . XXX

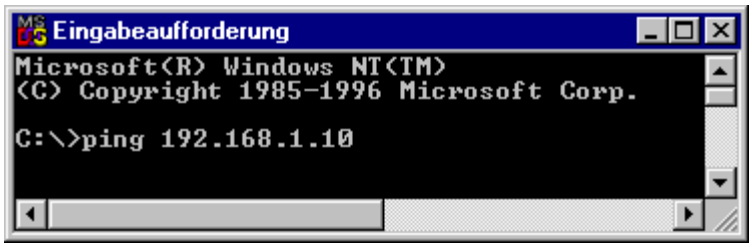


Abbildung 45: Beispiel für den Funktionstest eines Feldbusknotens

5. Drücken Sie die Taste **[Enter]**.
   Ihr Client-PC empfängt nun eine Antwort vom Feldbusknoten, die in der DOS-Eingabeaufforderung dargestellt wird.
   Falls stattdessen die Fehlermeldung „Zeitüberschreitung der Anforderung

(Timeout)“ erscheint, vergleichen Sie Ihre Eingaben nochmals mit der zugewiesenen IP-Adresse.

6. Bei erfolgreichem Test schließen Sie die DOS-Eingabeaufforderung.

Der Feldbusknoten ist jetzt für die Kommunikation vorbereitet.

8.4 Flash-Dateisystem vorbereiten

Die Vorbereitung des Flash-Dateisystems ist erforderlich, damit Sie alle weiteren Konfigurationen über das Web-Interface des Feldbuscontrollers durchführen können.

Im Auslieferungszustand ist das Flash-Dateisystem bereits vorbereitet.
Sollte jedoch bei Ihrem Feldbuscontroller das Flash-Dateisystem noch nicht initialisiert oder aufgrund eines Fehlers zerstört worden sein, müssen Sie dieses zunächst wie nachfolgend beschrieben extrahieren, um darauf zuzugreifen.

ACHTUNG

Kommunikationskabel nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 bzw. 750-923 nicht unter Spannung!
Der Feldbuskoppler/-controller muss dazu spannungsfrei sein!

Hinweis

Daten werden durch Formatieren gelöscht!
Beachten Sie, dass durch das Formatieren des Dateisystems alle Daten und gespeicherte Konfigurationen gelöscht werden.
Verwenden Sie diese Funktion nur dann, wenn das Flash-Dateisystem noch nicht initialisiert oder auf Grund eines Fehlers zerstört wurde.

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.

2. Schließen Sie das Kommunikationskabel 750-920 oder 750-923 bzw. den *Bluetooth*®-Adapter 750-921 an die Konfigurationsschnittstelle des Feldbuscontrollers und an Ihren PCs an.

3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot. Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information

Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**

5. Um das Dateisystem zu formatieren und nachfolgend die Webseiten in das Flash-Dateisystem zu extrahieren, wählen Sie in der oberen Menüleiste die Schaltfläche **Dateisystem zurücksetzen**.

Das Formatieren und Extrahieren ist beendet, sobald im unteren Statusfenster „Zurücksetzen des Dateisystems erfolgreich“ angezeigt wird.

Hinweis

Neustart des Feldbuskopplers/-controllers nach Zurücksetzen!
Damit der Webseiten nach dem Zurücksetzen angezeigt werden, muss der Feldbuskoppler/-controller neu gestartet werden.

8.5 Uhrzeit und Datum einstellen

Die Systemzeit des Feldbuscontrollers ermöglicht eine Datums- und Zeitangabe für Dateien im Flash Dateisystem.

Hinweis

Systemzeit wird im spannungslosen Zustand nicht weitergeführt!
Der Feldbuscontroller 750-852 verfügt über keinen Echtzeit-Uhrenbaustein. Aus diesem Grund wird die aktuelle Systemzeit im spannungslosen Zustand nicht weitergeführt!

Nach dem Einschalten der Betriebsspannung beginnt die Systemzeit bei 01.01.2000 00:00:00 Uhr.

Synchronisieren Sie die Systemzeit nach dem Einschalten der Betriebsspannung mit der aktuellen Rechnerzeit.

Um die Systemzeit zu synchronisieren gibt es zwei Möglichkeiten:

- Systemzeit mit den **WAGO-ETHERNET-Settings** synchronisieren
- Systemzeit über das **Web-based Management-System** synchronisieren

Systemzeit mit den WAGO-ETHERNET-Settings synchronisieren

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.
2. Schließen Sie das Kommunikationskabel 750-920 oder 750-923 bzw. den *Bluetooth*®-Adapter 750-921 an die Konfigurationsschnittstelle des Feldbuscontrollers und an Ihren PCs an.
3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.

Nach dem Einschalten der Betriebsspannung erfolgt die Initialisierung des Feldbuscontrollers. Dieser ermittelt die Busklemmenkonstellation und erstellt entsprechend das Prozessabbild.
Während des Hochlaufens blinkt die I/O-LED rot. Leuchtet nach kurzer Zeit die I/O-LED grün auf, ist der Feldbuscontroller betriebsbereit.

Tritt während des Hochlaufens ein Fehler auf, der mittels I/O-LED durch rotes Blinken als Fehlermeldung ausgegeben wird, werten Sie Fehlercode und -argument aus und beheben Sie den Fehler.

Information

Weitere Informationen zu der LED-Signalisierung
Entnehmen Sie die genaue Beschreibung für die Auswertung der angezeigten LED-Signale dem Kapitel „Diagnose“ > … > „LED-Signalisierung“.

4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**

5. Wählen Sie das Register **Datum und Uhrzeit**.

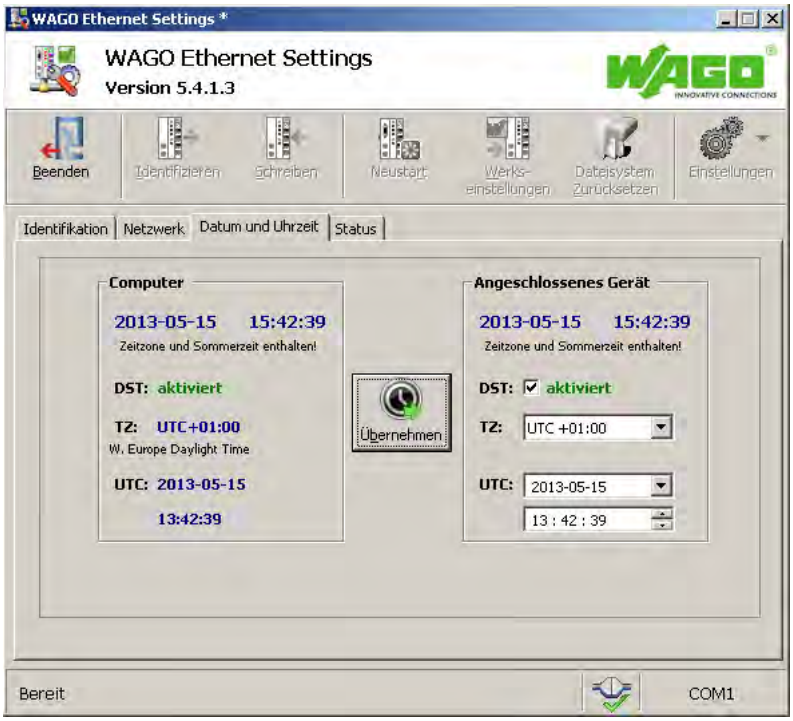


Abbildung 46: ETHERNET-Settings-Beispiel für die Zeitsynchronisation

6. Klicken Sie auf die Schaltfläche **[Übernehmen]**.

Systemzeit über das Web-based Management-System synchronisieren

1. Starten Sie einen Web-Browser (z.B. MS Internet-Explorer oder Mozilla) und geben Sie in der Adresszeile die IP-Adresse ein, die Sie Ihrem Feldbusknoten vergeben haben.

2. Bestätigen Sie mit **[Enter].**
   Die Startseite des Web-Interface wird aufgebaut.

3. Wählen Sie „Clock“ in der linken Menüleiste.

4. Geben Sie in der folgenden Abfrage Ihren Benutzernamen und das Passwort ein (Default: User = „admin“, Passwort = „wago“ oder: User = „user“, Passwort = „user“).
   Die HTML-Seite "Clock configuration" wird aufgebaut:

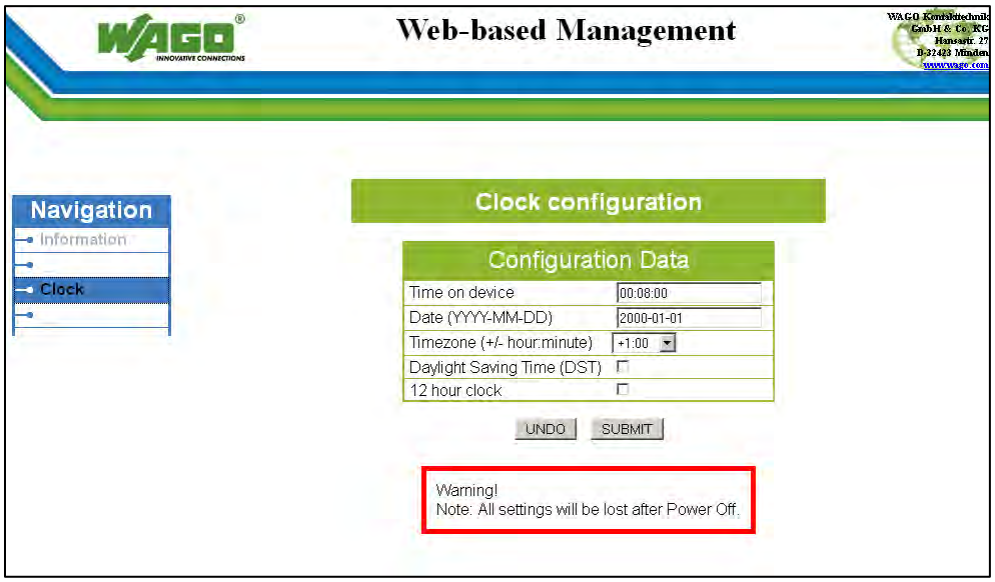


Abbildung 1: Beispiel WBM Clock configuration

5. Stellen Sie die Werte in den Feldern „Time on device“, „Date“ und „Timezone“ auf die entsprechend aktuellen Werte ein, und aktivieren Sie gegebenenfalls die Option "Daylight Saving Time (DST)".

6. Klicken Sie auf **[SUBMIT]**, um die Änderungen in Ihren Feldbusknoten zu übernehmen.

7. Damit die Einstellungen des Web-Interface wirksam werden, führen Sie einen Neustart des Feldbusknotens durch.

8.6 Werkseinstellungen wiederherstellen

Hinweis

Für vollständiges Löschen erst Dateisystem zurücksetzen!
Beachten Sie, dass Sie für das vollständige Löschen des Controllers zunächst das Dateisystem zurücksetzen. Nutzen Sie dazu das Programm **WAGO-ETHERNET-Settings**, wie nachfolgend beschrieben, und klicken Sie zuerst auf die Schaltfläche **[Dateisystem zurücksetzen]**. Anschließend stellen Sie die Werkseinstellungen wieder her.

Um die werksseitigen Einstellungen wiederherzustellen, gehen Sie wie folgt vor:

1. Schalten Sie die Betriebsspannung des Feldbuscontrollers aus.
2. Schließen Sie das Kommunikationskabel 750-920 oder 750-923 bzw. den *Bluetooth*®-Adapter 750-921 an die Konfigurationsschnittstelle des Feldbuscontrollers und an Ihren PCs an.
3. Schalten Sie die Betriebsspannung des Feldbuscontrollers wieder ein.
4. Starten Sie das Programm **WAGO-ETHERNET-Settings.**
5. Wählen Sie in der oberen Menüleiste die Schaltfläche **[Werkseinstellungen]** und bestätigen Sie die folgende Abfrage mit **[Ja]**.

Es wird automatisch ein Neustart des Feldbusknotens ausgeführt.
Der Start erfolgt mit den Werkseinstellungen.

9 PFC mit WAGO-I/O-*PRO* programmieren

Durch die IEC-61131-3-Programmierung kann das Gerät ETHERNET ECO Controller über die Funktionen eines Feldbuskopplers hinaus die Funktionalität einer SPS nutzen. Die Applikation gemäß IEC-61131-3 erstellen Sie mit dem Programmiertool WAGO-I/O-*PRO*.

Information

Kompatibilität mit der IEC-61131-3-Programmiersoftware!
Die Kompatibilität zwischen Ihrem Feldbuscontroller und der verwendeten IEC-61131-3-Programmiersoftware ist abhängig von der Firmware-Version des Feldbuscontrollers und der Version der Programmiersoftware.
Eine Referenzliste empfohlener Kombinationen finden Sie auf der Internetseite www.wago.com.
Verwenden Sie die Suchfunktion (Suchbegriff „Kompatibilitätshinweise“).

Hinweis

Option „CODESYS“ im Web-based Management-System muss aktiv sein!
Beachten Sie, dass als Voraussetzung für die IEC-61131-3-Programmierung des Controllers über ETHERNET das Kontrollkästchen **CODESYS** im Web-based Management-System auf der Seite „Port“ aktiviert sein muss (Default-Einstellung).
Alternativ können Sie Client-PC und Controller zur Programmierung jedoch auch seriell mit einem Programmierkabel verbinden.

Die Beschreibung der Programmierung mit WAGO-I/O-*PRO* ist nicht Bestandteil dieses Handbuchs. In den folgenden Kapiteln wird vielmehr auf wichtige Hinweise bei der Projekterstellung in der WAGO-I/O-*PRO* und auf spezielle Bausteine hingewiesen, die Sie explizit für die Programmierung des Controllers nutzen können.

Ferner wird beschrieben, wie Sie in WAGO-I/O-*PRO* einen geeigneten Kommunikationstreiber laden, ein IEC-61131-3-Programm auf den Controller übertragen und deren Abarbeitung starten.

Hinweis

Eine WAGO-I/O-*PRO*-/(CODESYS)-Instanz pro Zielsystem!
Beachten Sie, dass eine gleichzeitige Verbindung mehrerer WAGO-I/O-*PRO*-/(CODESYS)-Instanzen auf ein Zielsystem nicht möglich ist.

Hinweis

Namenskonventionen für WAGO-I/O-*PRO*-/(CODESYS)-Projekt beachten!
Beachten Sie, dass Sie für den Namen Ihres WAGO-I/O-*PRO*-/(CODESYS)-Projektes keine Sonderzeichen verwenden und diesen möglichst auf maximal 8 Zeichen beschränken.
Nur so ist sichergestellt, dass bei gleichzeitig aktivierter Online-Change-Funktion nicht für jedes Online-Change-Ereignis immer eine neue TxT-Datei erstellt wird, welche die Pfade und die Projekt-ID enthält, und somit zusätzlicher Speicher verbraucht wird. Bei korrekter Wahl des Dateinamens wird die TxT-Datei jedes Mal nur überschrieben und verbraucht keinen weiteren Speicher.

Information

Weitere Information
Eine detaillierte Beschreibung der Software-Bedienung entnehmen Sie dem Handbuch „WAGO-I/O-*PRO*“. Dieses finden Sie auf der Internetseite www.wago.com.

1. Starten Sie die Programmierumgebung unter **Startmenü \ Programme \ WAGO-I/O-*PRO***.

2. Legen Sie unter **Datei > Neu** ein neues Projekt an.

Sie erhalten ein Dialogfenster, in dem Sie das Zielsystem für die Programmierung einstellen.

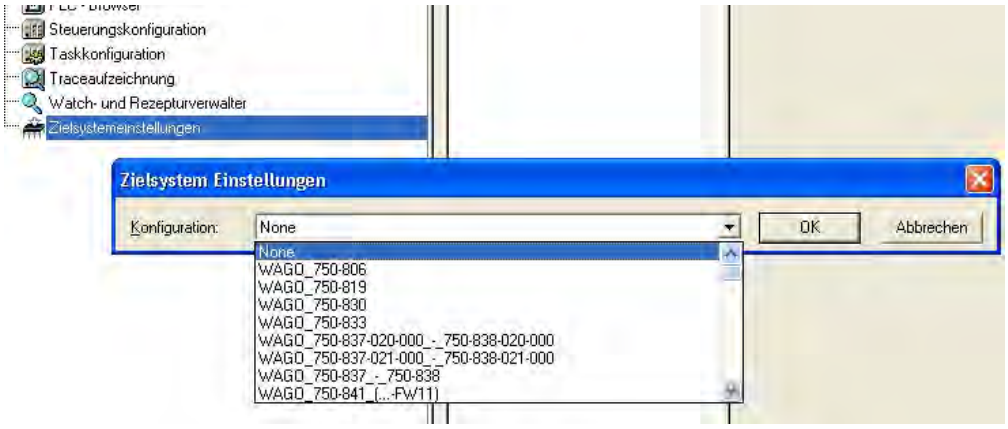


Abbildung 47: Dialogfenster Zielsystemeinstellungen

3. Wählen Sie in der Auswahlliste „Konfiguration:“ den entsprechenden Eintrag zu Ihrem Feldbuscontroller aus und bestätigen Sie mit **OK**.

4. Wählen Sie im folgenden Dialogfenster die Programmierart (AWL, KOP, FUP, AS, ST oder CFC) aus.

Damit Sie in Ihrem neuen Projekt definiert auf alle Busklemmendaten zugreifen können, ist zunächst die Busklemmenkonfiguration gemäß der vorhandenen Feldbusknoten-Hardware zusammenzustellen und in einer Konfigurationsdatei „EA-config.xml“ abzubilden.

In dieser Datei wird festgelegt, ob der Schreibzugriff auf die Busklemmen vom IEC-61131-3-Programm, von MODBUS/TCP oder von EtherNet/IP aus erfolgen darf.

Die Generierung der Datei kann, wie nachfolgend beschrieben, über die Konfiguration mit dem WAGO-I/O-Konfigurator erfolgen.

9.1 Feldbuscontroller mit dem I/O-Konfigurator konfigurieren

Der I/O-Konfigurator ist ein in der WAGO-I/O-*PRO* eingebundenes Plug-in zum Ermitteln von Adressen für die Busklemmen an einem Feldbuscontroller.

1. Wählen Sie im linken Bildschirmfenster der WAGO-I/O-*PRO*-Oberfläche die Registerkarte **Ressourcen**.

2. Um den I/O-Konfigurator zu starten, klicken Sie in der Baumstruktur doppelt auf **Steuerungskonfiguration**.

3. Erweitern Sie in der Baumstruktur den Zweig **Hardware configuration**.

4. Klicken Sie mit der rechten Maustaste auf den Eintrag **K-Bus** und wählen Sie im Kontextmenü **Bearbeiten**.

5. Klicken Sie im geöffneten Fenster „Konfiguration“ auf die Schaltfläche **Hinzufügen**, um das Modulauswahlfenster zu öffnen.

6. Wählen Sie die einzufügende Busklemme aus dem Modul-Katalog aus und hängen Sie diese mittels Schaltfläche **[>>]** und **[OK]** an das Ende der Klemmenbus-Struktur an.

7. Positionieren Sie alle notwendigen Busklemmen in der korrekten Reihenfolge, bis diese mit der Konfiguration des physikalischen Knotens übereinstimmen.

Projektieren Sie auf diese Weise die Baumstruktur in der Hardware-Konfiguration. Berücksichtigen Sie alle Busklemmen, die Daten liefern oder erwarten.

Greifen Sie online auf Ihren Feldbuscontroller zu, können Sie im Fenster „Konfiguration“ die Schaltfläche **[WAGO-I/O-*CHECK* starten und scannen]** verwenden, um den physikalisch angeschlossenen Feldbuscontroller mit angereihten Busklemmen einzulesen und alle Komponenten anzuzeigen.

Hinweis

Der Klemmenbusaufbau im WAGO-I/O-Konfigurator muss mit dem physikalischen Knotenaufbau übereinstimmen!
Die Anzahl der Busklemmen, die Daten liefern oder erwarten, muss unbedingt mit der vorhandenen Hardware übereinstimmen (ausgenommen sind z. B. Potentialeinspeise-, Vervielfältigungs- und Endklemmen). Die Anzahl der Ein-/Ausgangsbits oder -bytes der einzelnen angeschalteten Busklemmen entnehmen Sie den entsprechenden Beschreibungen der Busklemmen.

Information

Weitere Information

Um das Datenblatt einer Busklemme zu öffnen, klicken Sie im Fenster „Konfiguration“ auf die betreffende Busklemme und drücken die Schaltfläche **[Datenblatt]**. Das Datenblatt wird in einem neuen Fenster angezeigt.
Alle aktuellen Datenblätter finden Sie auf der Internetseite http://www.wago.com unter Dokumentation.

8. Mit der Schaltfläche **[OK]** übernehmen Sie die Knotenkonfiguration und schließen den Dialog.

Die Adressen der Steuerungskonfiguration werden neu berechnet und die Baumstruktur der Steuerungskonfiguration aktualisiert.

Ändern Sie nun gegebenenfalls die gewünschte Zugriffsberechtigung für einzelne Busklemmen, wenn auf diese über einen Feldbus (z. B. MODBUS TCP/IP oder EtherNet/IP) zugegriffen werden soll. Zunächst ist für jede eingefügte Busklemme der Schreibzugriff vom PLC aus festgelegt. Um dieses zu ändern, führen Sie folgende Schritte aus:

9. Klicken Sie im Dialogfenster „Konfiguration“ auf das Register „PA-Zuweisung“.

10. Bestimmen Sie für jede einzelne Busklemme, von wo aus der Zugriff auf die Busklemmendaten erfolgen soll.

Hierbei haben Sie in der Spalte „PA-Zuordnung“ folgende Auswahl:

- PLC, PLC (Standardeinstellung) - Zugriff vom PFC aus
- fieldbus 1 - Zugriff von MODBUS/TCP aus
- fieldbus 2 - Zugriff von Ethernet/IP aus

Nach Fertigstellung der Zuordnung, können Sie mit der IEC-61131-3-Programmierung beginnen.

Wenn Sie das Projekt übersetzen (Menü **Projekt** > **Übersetzen**/**Alles übersetzen**) und in den Feldbuscontroller laden, wird in dem Feldbuscontroller automatisch eine Konfigurationsdatei „EA-config.xml“ generiert und abgelegt.

Hinweis

Bei direktem Schreiben über MODBUS an eine Hardwareadresse „MODBUS TCP/UDP, fieldbus1“ einstellen!

Wenn Sie über MODBUS direkt auf eine Hardware-Adresse schreiben wollen, stellen Sie den Zugriff über „MODBUS TCP/UDP, fieldbus1“ ein. Ansonsten sind die Busklemmen der SPS zugeordnet und es ist kein Schreiben von außen möglich.

Information

Weitere Information

Eine detaillierte Beschreibung zur Bedienung der Software WAGO-I/O-*PRO* und des I/O-Konfigurators finden Sie auch in der Online-Hilfe zur WAGO-I/O-*PRO*.

9.1.1 Feldbuscontroller mit der Datei „EA-config.xml“ konfigurieren

Sie können die Konfigurationsdatei „EA-config.xml“ neben der automatischen Generierung in WAGO-I/O-*PRO* auch manuell anlegen und ändern.

Die Datei legen Sie per FTP in dem Verzeichnis „/etc“ auf dem Feldbuscontroller ab.

Im Folgenden wird die Konfiguration des Feldbuscontrollers mittels der Konfigurationsdatei „EA-config.xml“ beschrieben.

Hinweis

Konfigurationseinträge in WAGO-I/O-*PRO* überschreiben „EA-config.xml“ bei Download!
Wenn Sie die Busklemmenzuordnung direkt mittels der im Feldbus-controller gespeicherten Datei „EA-config.xml“ vornehmen, dürfen Sie zuvor keine Konfigurationseinträge in der WAGO-I/O-*PRO* speichern, da die Datei durch die Einträge in der WAGO-I/O-*PRO* bei jedem Download überschrieben wird.

1. Stellen Sie eine Verbindung via FTP zu Ihrem Feldbuscontroller her. Dazu verwenden Sie ein FTP-Programm oder geben in Ihrem Browser folgende Zeile ein:

   ftp://[IP-Adresse des Feldbuscontrollers], z. B. ftp://192.168.1.201

2. Anschließend geben Sie den Benutzernamen **admin** sowie das Kennwort **wago** ein.

Die Datei „EA-config.xml“ liegt im Ordner „/etc“ auf dem Feldbuscontroller.

3. Kopieren Sie die Datei in ein lokales Verzeichnis auf Ihrem PC und öffnen Sie diese in einem beliebigen Editor (z. B. „WordPad“).

Die folgende Syntax ist bereits in der Datei vorbereitet:

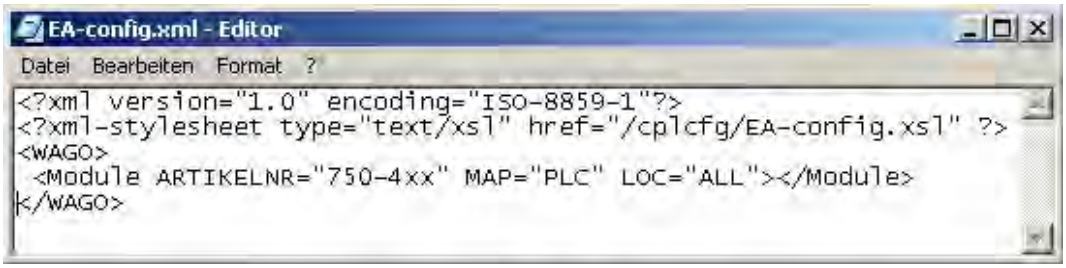


Abbildung 48: Konfigurationsdatei „EA-config.xml“

Die vierte Zeile enthält die notwendigen Informationen für die erste Busklemme. Der Eintrag MAP=„PLC“ weist dem IEC-61131-3-Programm die Schreibzugriffsrechte für das erste Modul zu.

4. Wenn Sie den Zugriff über MODBUS/TCP ermöglichen wollen, ersetzen Sie „PLC“ durch „FB1“ und für den Zugriff von Ethernet/IP aus durch „FB2“:

   <Module BESTELLNR=“ “ MAP=“**PLC**“ LOC=“ALL“> </Module>

   ↓

   <Module BESTELLNR=“ “ MAP=“**FB1**“ LOC=“ALL“> </Module>

5. Um weitere Busklemmen hinzuzufügen, ergänzen Sie unter der vierten Zeile für jede einzelne Ihrer montierten Busklemmen eine neue Zeile in derselben Syntax und setzen Sie entsprechende Zugriffsberechtigungen.

Hinweis

Anzahl der Zeileneinträge gleich der Anzahl verwendeter Busklemmen!
Die Anzahl der Zeileneinträge muss unbedingt mit der Anzahl der vorhandenen Busklemmen in der Hardware-Konstellation übereinstimmen.

6. Speichern Sie die Datei und laden Sie diese wieder über den FTP-Client in das Dateisystem des Controllers.

Im Anschluss daran beginnen Sie mit der IEC-61131-3-Programmierung.

Information

Weitere Information
Eine detaillierte Beschreibung der Software-Bedienung entnehmen Sie dem Handbuch WAGO-I/O-*PRO*. Dieses finden Sie im Internet unter: www.wago.com → Downloads → Zusätzliche Dokumentation und Information für Automatisierungsprodukte → WAGO-Software → WAGO-I/O-*PRO* / CODESYS > 759-333

9.2 ETHERNET-Bibliotheken für WAGO-I/O-*PRO*

Für unterschiedliche IEC-61131-3-Programmieraufgaben stehen Ihnen in WAGO-I/O-*PRO* verschiedene Bibliotheken zur Verfügung. Diese enthalten universell einsetzbare Funktionsbausteine und können somit Ihre Programmerstellung erleichtern und beschleunigen.

Nach dem Einbinden der Bibliotheken können Sie auf Funktionsbausteine, Funktionen und Datentypen zugreifen, die Sie genauso benutzen können, wie selbstdefinierte.

Information **Weitere Information**

Sie finden alle Bibliotheken auf der Installations-CD zur Software WAGO-I/O-*PRO* oder im Internet unter http://www.wago.com.

Nachfolgende Bibliotheken stehen Ihnen spezifisch für ETHERNET-Projekte mit WAGO-I/O-*PRO* zur Verfügung.

Tabelle 34: ETHERNET-Bibliotheken für WAGO-I/O-*PRO*

Bibliothek	Beschreibung
Ethernet.lib	Funktionsbausteine zur Kommunikation via ETHERNET
WAGOLibEthernet_01.lib	Funktionsbausteine zur Herstellung einer Verbindung mit einem Remote-Server oder Client-PC (über TCP) und zum Datenaustausch mit jedem möglichen UDP-Server oder Client-PC (über UDP)
WAGOLibModbus_IP_01.lib	Funktionsbausteine zur Herstellung einer Verbindung mit einem oder mehreren Slaves
ModbusEthernet_04.lib	Funktionsbausteine für den Datenaustausch mit mehreren MODBUS/TCP/UDP-Slaves Stellt außerdem einen MODBUS-Server zur Verfügung, welcher die MODBUS-Dienste auf einem Word-Array abbildet.
SysLibSockets.lib	Funktionsbausteine für den Zugriff auf Sockets zur Kommunikation über TCP/IP und UDP
WagoLibSockets.lib	Funktionsbausteine für den Zugriff auf Sockets zur Kommunikation über TCP/IP und UDP Enthält im Gegensatz zu SysLibSockets.lib weitere Funktionen.
Mail_02.lib	Funktionsbausteine zum Versenden von Emails
WAGOLibMail_02.lib	Funktionsbausteine zum Versenden von Emails
WagoLibSntp.lib	Funktionsbausteine zur Einstellung und Verwendung des Simple-Network-Time-Protocols (SNTP)
WagoLibFtp.lib	Funktionsbausteine zur Einstellung und Verwendung des File-Transfer-Protocols (FTP)
WAGOLibTerminalDiag.lib	Funktionsbausteine zur Ausgabe von Modul-, Kanal- und Diagnosedaten von Klemmen, die Diagnosedaten bereitstellen

Information **Weitere Information**

Eine detaillierte Beschreibung der Bausteine und der Software-Bedienung entnehmen Sie der Online-Hilfe von WAGO-I/O-*PRO* oder dem Handbuch WAGO-I/O-*PRO* auf der Internetseite http://www.wago.com.

9.3 Einschränkungen im Funktionsumfang

Die Basis von WAGO-I/O-*PRO*, das Standard Programmiersystem CODESYS von 3S, besitzt eine integrierte Visualisierung. Diese Visualisierung kann je nach Zielsystem in den Varianten „HMI“, „TargetVisu“ und „WebVisu“ genutzt werden.

Der Feldbuscontroller 750-852 unterstützt nur die Ablaufvariante „HMI“.

In dieser Variante stehen komplexe Visualisierungsobjekte, wie z. B. „Alarm“ und „Trend“ zur Verfügung. Dieses gilt z. B. für das Versenden von Mails als Reaktion auf einen Alarm oder für die Navigation durch historische Trenddaten sowie deren Erzeugung. Die Ablaufvariante „WebVisu“ steht nicht zur Verfügung.

Über die Visualisierung hinaus gelten folgende globale Einschränkungen für das Programmiersystem:

Dateisystem (2,0 MB):
Die Gesamtgröße von SPS-Programm, Logdateien, Konfigurationsdateien usw. muss in das Dateisystem passen.
Die Größe des freien Speicherplatzes liefert der PLC-Browser auf das Kommando „fds“ (FreeDiscSpace).

Bausteinanzahl (standardmäßig 1023):
Die Gesamtgröße des SPS-Programmes wird unter anderem durch die maximale Bausteinanzahl bestimmt. Dieser Wert ist in den Zielsystemeinstellungen konfigurierbar.

Rechenleistung/Prozessorzeit:
Das Gerät ETHERNET ECO Controller basiert auf einem Echtzeitbetriebssystem mit präemptivem Multitasking. Dabei verdrängen hochpriore Prozesse, wie z. B. das SPS-Programm, niederpriore Prozesse, wie z. B. Webserver.
Einen Überblick über die realen Ausführungszeiten aller Tasks in WAGO-I/O-*PRO* liefert der PLC-Browser auf das Kommando „tsk“.
Werden in einem SPS-Programm Betriebsystemfunktionen wie z. B. für das Handling von Sockets oder dem Dateisystem verwendet, werden diese Ausführungszeiten vom Kommando „tsk“ nicht berücksichtigt.

Zähler CTU:
Der Zähler CTU arbeitet im Wertebereich von 0 bis 32767.

Hinweis

Maximale Schreibzyklen des EEPROM beachten!
Feldbuskoppler/-controller speichern einige Informationen wie IP-Adresse und IP-Parameter im EEPROM, damit diese nach einem Neustart verfügbar sind. Die Speicherzyklen eines EEPROM sind generell begrenzt. Oberhalb einer Grenze von etwa 1 Million Schreibzyklen kann der Speicher nicht mehr zugesichert werden. Ein defektes EEPROM macht sich erst bei einem Neustart durch Software-Reset oder Power-ON bemerkbar.
Der Feldbuskoppler/-controller startet dann aufgrund einer fehlerhaften Checksumme im EEPROM immer wieder mit den Default-Parametern.

Folgende Funktionen verwenden das EEPROM:

- **WAGO-I/O-PRO**
- **EthernetLib** SetNetworkConfig
  SetVariables
- **MODBUS**
  - Register 0x1035 Time Offset
  - Register 0x100B Watchdog Parameter
  - Register 0x1028 Netzwerk Konfiguration
  - Register 0x1036 Daylight Saving
  - Register 0x1037 Modbus Response Delay
  - Register 0x2035 PI-Parameter
  - Register 0x2043 Default Konfiguration
- **Ethernet/IP**
  - Klasse 0xF5
  - Klasse 0xF6
  - Klasse 0x64
- **Parameterzuweisungen**
  - **BootP** Neue Parameter
  - **DHCP** Neue Parameter
  - **WAGO-MIB** Schreibzugriffe

9.4 Generelle Hinweise zu den IEC-Tasks

Beachten Sie bei der Programmierung Ihrer IEC-Tasks die folgenden Hinweise:

Hinweis

Unterschiedliche Prioritäten für IEC-Tasks verwenden!
Versehen Sie IEC-Tasks (in WAGO-I/O-*PRO* unter Register **Ressourcen** > **Task-Konfiguration**) mit unterschiedlichen Prioritäten, ansonsten kommt es beim Übersetzen des Anwenderprogramms zu einer Fehlermeldung.

Unterbrechung von IEC-Tasks durch Tasks höherer Priorität möglich!
Eine laufende IEC-Task kann durch Tasks mit höherer Priorität in ihrer Ausführung unterbrochen werden. Erst wenn keine Task mit höherer Priorität mehr zur Ausführung ansteht, wird die Ausführung der unterbrochenen Task wieder aufgenommen

Verfälschung von Variablen in überlappenden Bereichen des Prozessabbildes!
Verwenden mehrere IEC-Tasks Ein-/Ausgangsvariablen mit gleichen oder überlappenden Adressen im Prozessabbild, können sich die Werte der Ein-/Ausgangsvariablen während der Ausführung der IEC-Task ändern!

Wartezeit bei freilaufenden Tasks beachten!
Freilaufende Tasks werden nach jedem Task-Zyklus für die Hälfte der Zeit angehalten, welche die jeweilige Task selbst benötigt (mind. 1 ms).
Danach beginnt die erneute Ausführung.

Beispiel: 1. Task 4 ms → Wartezeit 2 ms
2. Task 2 ms → Wartezeit 1 ms

Default-Task wird standardmäßig angelegt!
Auch wenn Sie in WAGO-I/O-*PRO* im Register **Ressourcen** > **Task-Konfiguration** keine Task anlegen, wird automatisch eine freilaufende „DefaultTask“ erstellt.
Achten Sie darauf, für Ihre Task nicht denselben Namen zu verwenden.

Watchdog-Empfindlichkeit für zyklische Tasks beachten!
Die Watchdog-Empfindlichkeit gibt an, bei wie vielen Überschreitungen der Watchdog-Zeit ein Ereignis ausgelöst wird. Die Empfindlichkeit stellen Sie in WAGO-I/O-*PRO* unter Register **Ressourcen** > **Task-Konfiguration** für zyklische Tasks ein. Eine Empfindlichkeit von 0 oder 1 ist gleichbedeutend und bewirkt, dass bei einmaliger Überschreitung der Watchdog-Zeit das Watchdog-Ereignis ausgelöst wird.
Bei einer Empfindlichkeit von 2, wird beispielsweise ein Watchdog-Ereignis ausgelöst, wenn in zwei aufeinanderfolgenden Taskzyklen die Watchdog-Zeit überschritten wird.

Für zyklische Tasks mit aktiviertem Watchdog zur Zykluszeitüberwachung gilt:

Hinweis

Hinweise zur Einstellung des Watchdogs!
Für jede angelegte Task kann ein Watchdog aktiviert werden, der die Abarbeitungszeit der jeweiligen Task überwacht.
Überschreitet die Tasklaufzeit die angegebene Watchdog-Zeit (z. B. t#200 ms), dann ist der Watchdog-Fall eingetreten.
Das Laufzeitsystem stoppt das IEC-Programm und meldet einen Fehler.

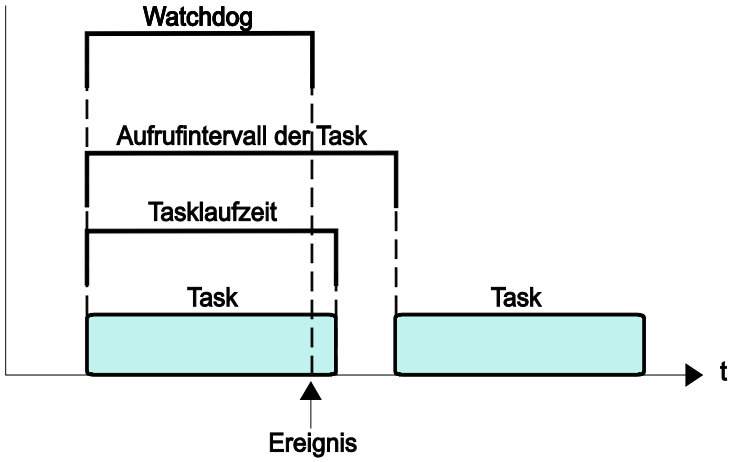


Abbildung 49: Watchdog-Laufzeit kleiner als Tasklaufzeit

Ist die eingestellte Watchdog-Zeit größer als das Aufrufintervall der Task, wird zu jedem Aufrufintervall der Watchdog neu gestartet.

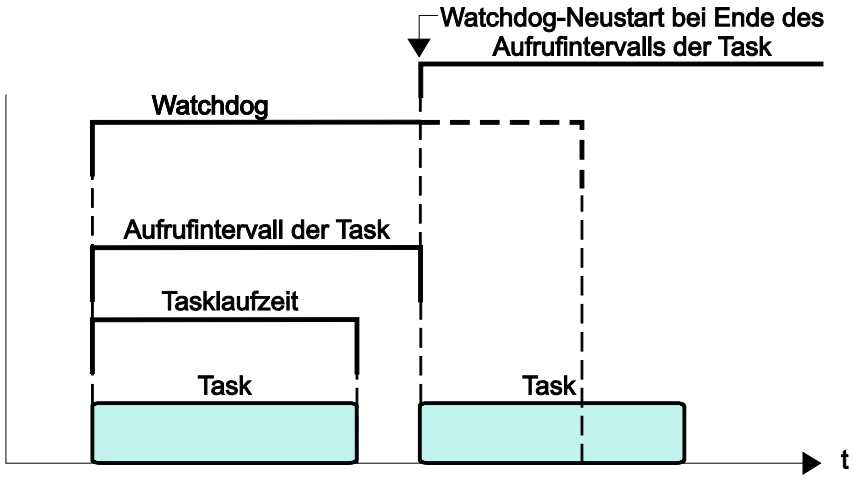


Abbildung 50: Watchdog-Laufzeit größer als Task-Aufrufintervall

Für zyklische Tasks gilt:

Hinweis

Keine zyklischen Tasks mit Aufrufintervall > 30 min. möglich!
Es sind keine zyklischen Tasks mit einem Aufrufintervall größer als 30 Minuten möglich.

9.4.1 Ablaufschema einer IEC-Task

1. Systemzeit ermitteln (tStart)
2. Wenn seit dem letzten Schreiben der Ausgänge noch kein vollständiger Klemmenbuszyklus gefahren wurde
   → Auf das Ende des nächsten Klemmenbuszyklus warten
3. Eingänge lesen und Ausgänge aus dem Prozessabbild zurücklesen
4. Wenn das Anwenderprogramm gestartet wurde
   → Programmcodes dieser Task ausführen
5. Ausgänge in das Prozessabbild schreiben
6. Systemzeit ermitteln (tEnd)
   → tEnd - tStart = Laufzeit der IEC-Task

9.4.2 Die wichtigsten Task-Prioritäten im Überblick

Tabelle 35: Task-Abarbeitung

Task	Wichtigkeit der Abarbeitung
Klemmenbus-Task, Feldbus-Task	vorrangig vor allen anderen
Normale Task	nach den Klemmenbus- und Feldbus-Tasks
PLC-Comm-Task	nach den Normalen Tasks
Background-Task	nach den PLC-Comm-Tasks

Klemmenbus-Task/Feldbus-Task (Intern)
Bei der Klemmenbus-Task handelt es sich um eine interne Task, die zyklisch das Prozessabbild mit den Ein-/Ausgangsdaten der Busklemmen abgleicht.
Die Feldbus-Tasks laufen ereignisgesteuert und nehmen lediglich Rechenzeit in Anspruch, wenn über den Feldbus kommuniziert wird (MODBUS/Ethernet/IP).

Normale Task (in CODESYS einstellbare IEC-Task Priorität 1)
IEC-Tasks mit dieser Priorität können durch die Klemmenbus-Task unterbrochen werden. Deshalb muss die gesteckte Klemmenkonfiguration und die Kommunikation über den Feldbus bei aktiviertem Watchdog für das Task-Aufrufintervall berücksichtigt werden.

PLC-Comm-Task (Intern)
Die PLC-Comm-Task ist im eingeloggten Zustand aktiv und übernimmt die Kommunikation mit dem Gateway der WAGO-I/O-*PRO*.

Background-Task (in CODESYS einstellbare IEC-Task Priorität 2)
Alle internen Tasks haben eine höhere Priorität als die IEC-Background-Task. Von daher eignet sich diese Task besonders, um zeitintensive und zeitunkritische Aufgaben durchzuführen.

Hinweis

Maximal Anzahl IEC-Tasks!
Beachten Sie, dass der Feldbuscontroller insgesamt 30 Tasks (Task-ID 0 … 29) unterstützt, wobei zwei Tasks (Task-ID 0 … 1) von dem Laufzeitsystem in der FW verwendet werden. Es verbleiben Ihnen somit 28 Tasks (Task-ID 2 … 29) für Ihr Anwenderprogramm. Der Überschuss an Tasks hat als Task-ID ein ungültiges Handle (0xFFFFFFFF).

Information

Weitere Information
Eine detaillierte Beschreibung zu dem Programmiertool WAGO-I/O-*PRO* entnehmen Sie dem Handbuch WAGO-I/O-*PRO* im Internet unter http://www.wago.com.

9.5 Systemereignisse

Anstelle einer Task kann auch ein Systemereignis (Event) einen Projektbaustein zur Abarbeitung aufrufen.

Die dazu verwendbaren Systemereignisse sind zielsystemabhängig. Sie setzen sich zusammen aus der Liste der unterstützten Standardsystemereignisse der Steuerung und eventuell hinzugefügten herstellerspezifischen Ereignissen.

Mögliche Ereignisse sind z. B. „Stop“, „Start“, „Online Change“.

Die vollständige Liste aller Systemereignisse wird in WAGO-I/O-*PRO* aufgeführt.

9.5.1 Systemereignisse aktivieren/deaktivieren

1. Öffnen Sie in WAGO-I/O-*PRO* das Register **Ressourcen** > **Task-Konfiguration** > **Systemereignisse** (siehe folgende Abbildung).

2. Damit ein Baustein durch ein Ereignis aufgerufen werden kann, aktivieren Sie die gewünschten Einträge durch Setzen von Haken in die betreffenden Kontrollkästchen.

3. Deaktivieren Sie Kontrollkästchen, indem Sie die Haken mit einem Mausklick entfernen.

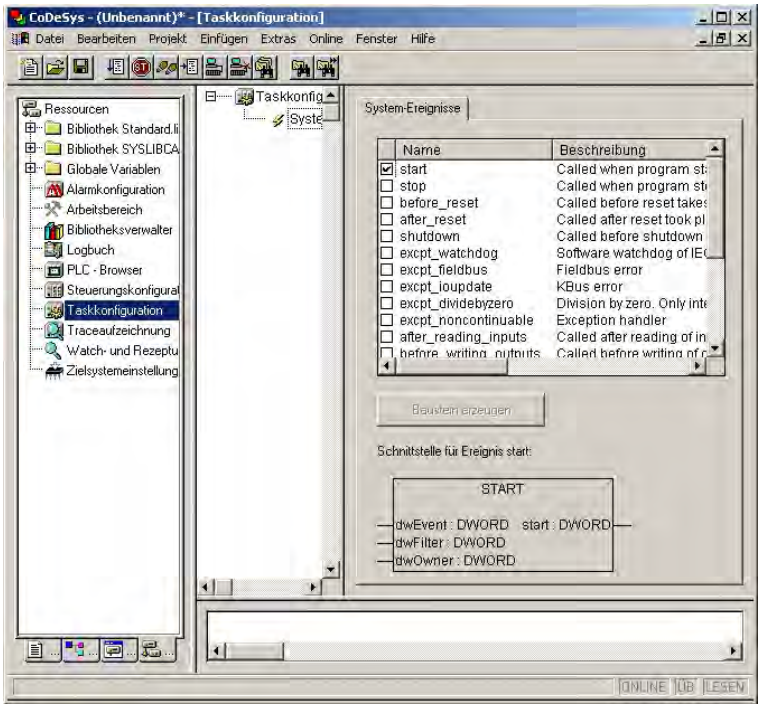


Abbildung 51: Systemereignisse aktivieren/deaktivieren

Hinweis

Ausschließlich Globale Variablen und Funktionen (Fun) verwenden!
Beachten Sie, dass der Aufruf von Funktionsblöcken (FB) wie z.B. Timer-Bausteine oder der Aufruf von blockierenden Funktionen wie Fileoperationen zu einem Fehlverhalten des Systems führen kann. Verwenden Sie deshalb in Systemereignissen ausschließlich Globale Variablen und Funktionen (Fun).

Information

Weitere Information
Die Zuordnung der Systemereignisse zu dem jeweils aufzurufenden Funktionsbaustein finden Sie detailliert beschrieben in dem Handbuch zum Programmiertool WAGO-I/O-*PRO* im Internet unter http://www.wago.com.

9.6 IEC-Programm auf den Controller übertragen

Sie können eine erstellte IEC-61131-Applikation auf zwei Arten von Ihrem PC auf den Controller übertragen (siehe folgende Kapitel):

- mittels serieller Schnittstelle direkt übertragen
- mittels Feldbus übertragen

Für die Übertragung sind geeignete Kommunikationstreiber erforderlich, welche Sie in WAGO-I/O-*PRO* laden und konfigurieren.

Hinweis

Kommunikationsparameter des Treibers anpassen!
Achten Sie bei der Auswahl des gewünschten Treibers auf die richtigen Einstellungen und Anpassungen der Kommunikationsparameter (siehe nachfolgende Beschreibung).

Hinweis

„Reset“ und „Start“ zum Setzen der physikalischen Ausgänge notwendig!
Die Initialisierungswerte für die physikalischen Ausgänge werden nicht direkt nach dem Download gesetzt. Wählen sie in der Menüleiste der WAGO-I/O-*PRO* **Online** > **Reset** und nachfolgend **Online** > **Start** zum Setzen der Werte.

Hinweis

Applikation vor dem Erzeugen großer Bootprojekte stoppen!
Stoppen Sie vor dem Erzeugen eines sehr großen Bootprojektes die WAGO-I/O-*PRO*-Applikation mittels **Online** > **Stop**, da es sonst zu einem Stoppen des Klemmenbusses kommen kann. Nach dem Erzeugen des Bootprojektes können Sie die Applikation wieder starten.

Hinweis

Handling persistenter Daten beeinflusst den Programmstart!
In Abhängigkeit von Variablentyp, Anzahl und Größe der persistenten Daten sowie deren Kombination, z. B. in Funktionsbausteinen, kann das Handling mit persistenten Daten den Programmstart durch eine verlängerte Initialisierungsphase verzögern.

Information

Weitere Informationen
Die folgende Beschreibung dient dem schnellen Einstieg. Die Installation fehlender Kommunikationstreiber sowie die detaillierte Software-Bedienung entnehmen Sie dem Handbuch WAGO-I/O-*PRO* im Internet unter http://www.wago.com.

9.6.1 Applikation mittels serieller Service-Schnittstelle übertragen

Hinweis

Stellung des Betriebsartenschalters bei Zugriff auf Controller beachten!
Für den Zugriff auf den Feldbuscontroller muss der Betriebsartenschalter, der sich hinter der Abdeck-Klappe des Feldbuscontrollers neben der Service-Schnittstelle befindet, in der mittleren oder in der oberen Stellung sein.

Um eine physikalische Verbindung über die serielle Service-Schnittstelle herzustellen, verwenden Sie das WAGO-Kommunikationskabel. Dieses ist im Lieferumfang der Programmiersoftware WAGO-I/O-*PRO* (Art.-Nr.: 759-333) enthalten oder kann als Zubehör über die Bestell-Nr.: 750-920 bezogen werden.

ACHTUNG

Kommunikationskabel nicht unter Spannung stecken!
Um Schäden an der Kommunikationsschnittstelle zu vermeiden, stecken und ziehen Sie das Kommunikationskabel 750-920 bzw. 750-923 nicht unter Spannung!
Der Feldbuscontroller muss dazu spannungsfrei sein!

1. Kontrollieren Sie, ob sich der Betriebsartenschalter in der mittleren oder in der oberen Stellung befindet.
   Sollte dieses nicht der Fall sein, bringen Sie den Betriebsartenschalter in die mittlere oder obere Stellung.

2. Verbinden Sie über das WAGO-Kommunikationskabel eine COM-Schnittstelle Ihres PCs mit der seriellen Service-Schnittstelle des Feldbuscontrollers.

Für die serielle Datenübertragung ist ein Kommunikationstreiber erforderlich. Dieser Treiber und seine Parametrierung wird in WAGO-I/O-*PRO* in dem Dialog „Kommunikationsparameter“ eingetragen:

3. Starten Sie die Software WAGO-I/O-*PRO* unter **Startmenü > Programme > WAGO-Software > WAGO-I/O-PRO**.

4. Wählen Sie im Menü **Online** den Unterpunkt **Kommunikationsparameter** aus.

Der Dialog „Kommunikationsparameter“ öffnet sich. Auf der linken Seite des Dialoges werden die Kanäle der aktuell verbundenen Gateway-Server und darunter die bereits installierten Kommunikationstreiber angezeigt.
In der Grundeinstellung sind in diesem Dialog noch keine Einträge vorhanden.

5. Klicken Sie auf **Neu...**, um eine neue Verbindung herzustellen und vergeben Sie einen Namen, z. B. RS-232-Verbindung.

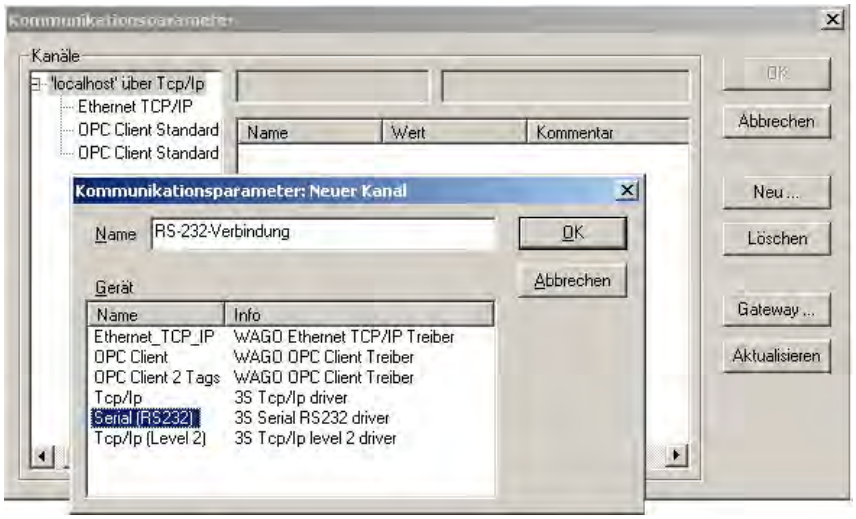


Abbildung 52: Dialogfenster „Kommunikationsparameter“, Erstellen einer neuen Verbindung

6. Markieren Sie in dem Auswahlfenster auf der rechten Seite des Dialogs den gewünschten Treiber **Serial (RS-232) – 3S Serial RS-232 driver**, um die serielle Verbindung zwischen PC und Feldbuscontroller zu konfigurieren.

In dem mittleren Fenster des Dialogs sind die folgenden Standardeinträge vorhanden:

- Port: COM1
- Baudrate: 19200
- Parity: Even
- Stop-bits: 1
- Motorola byteorder: No

7. Ändern Sie gegebenenfalls die Einträge entsprechend der obigen Werte ab, indem Sie auf den jeweiligen Wert klicken und diesen editieren.

8. Bestätigen Sie mit **OK**

Die RS-232-Schnittstelle ist nun für das Übertragen der Applikation konfiguriert.

Hinweis

Zur Programmierung des Feldbuscontrollers mit Administrator-kennwort anmelden!
Haben Sie auf der Seite „Security“ des Web-based Management-Systems den Passwortschutz für Port 2455 aktiviert, müssen Sie sich in WAGO-I/O-*PRO* im Menü **Online** > **Einloggen** anmelden, um Programmierzugriff auf den Feldbuscontroller zu erhalten (Standardkennwort „wago“).

Eingabe Steuerungs-Passwort
Der Online-Zugriff auf die Steuerung ist mit einem Passwort geschützt. Bitte geben Sie cas Passwort ein:
OK
Abbrechen

Abbildung 53: Anmeldung für Programmierzugriff

9. Um eine Verbindung mit dem Feldbuscontroller aufzubauen, klicken Sie im Menü **Online** auf **Einloggen**.

Durch das Einloggen wird der Online-Modus zum Feldbuscontroller eingeschaltet und die Kommunikationsparameter sind nicht mehr aufrufbar.

Sofern noch kein Programm im Feldbuscontroller vorhanden ist, erscheint ein Fenster mit der Abfrage, ob das Programm geladen werden soll.

10. Um das aktuelle Programm zu laden, bestätigen Sie mit **Ja**.

11. Klicken Sie im Menü **Online** auf **Bootprojekt erzeugen**.

Auf diese Weise wird Ihr kompiliertes Projekt auch ausgeführt, wenn Sie den Feldbuscontroller neu starten oder wenn es einen Spannungsausfall gibt.

12. Wenn das Programm geladen ist, starten Sie die Programmabarbeitung über das Menü **Online** und den Menüpunkt **Start**.

Dieser Befehl startet die Abarbeitung Ihres Programms in der Steuerung bzw. in der Simulation.

Am rechten Ende der Statusleiste wird „ONLINE“ und „LÄUFT“ angezeigt.

13. Um den Online-Betrieb zu beenden, klicken Sie im Menü **Online** auf den Menüpunkt **Ausloggen**.

9.6.2 Applikation mittels Feldbus via ETHERNET übertragen

Die physikalische Verbindung zwischen PC und Feldbuscontroller erfolgt über das Feldbuskabel.
Für die Datenübertragung ist ein geeigneter Kommunikationstreiber erforderlich. Den Treiber und seine Parameter tragen Sie in WAGO-I/O-*PRO* im Dialog „Kommunikationsparameter“ ein:

Hinweis

Feldbuscontroller benötigt IP-Adresse für den Zugriff!
Damit Sie auf den Feldbuscontroller zugreifen können, benötigt der Feldbuscontroller eine IP-Adresse. Der Betriebsartenschalter, der sich hinter der Abdeck-Klappe des Feldbuscontrollers neben der Service-Schnittestelle befindet, muss in der mittleren oder in der oberen Stellung sein.

1. Starten Sie die Software WAGO-I/O-*PRO* unter **Startmenü > Programme > WAGO-Software > WAGO-I/O-PRO.**

2. Wählen Sie im Menü **Online** den Unterpunkt **Kommunikationsparameter** aus.

Der Dialog „Kommunikationsparameter“ öffnet sich. Auf der linken Seite des Dialoges werden die Kanäle der aktuell verbundenen Gateway-Server und darunter die bereits installierten Kommunikationstreiber angezeigt.
In der Grundeinstellung sind in diesem Dialog noch keine Einträge vorhanden.

3. Klicken Sie auf **Neu...**, um eine neue Verbindung herzustellen und vergeben Sie einen Namen, z. B. TcpIp-Verbindung.

4. Markieren Sie in dem Auswahlfenster auf der rechten Seite des Dialogs den gewünschten TCP/IP-Treiber, um die Verbindung zwischen PC und Feldbuscontroller via ETHERNET zu konfigurieren.
   Verwenden Sie die neue Treiber-Version **Tcp/Ip** (3S Tcp/Ip driver).

In dem mittleren Fenster des Dialogs sind die folgenden Standardeinträge vorhanden:

- Adresse: IP-Adresse des Feldbuscontrollers
- Port: 2455
- Motorolabyteorder: No
- Debug Stufe: 16#0000

5. Ändern Sie gegebenenfalls die Einträge entsprechend der obigen Werte ab, indem Sie auf den jeweiligen Wert klicken und diesen editieren.

6. Bestätigen Sie mit **OK**.

Die TCP/IP-Schnittstelle ist nun für das Übertragen der Applikation konfiguriert.

7. Um eine Verbindung mit dem Feldbuscontroller aufzubauen, klicken Sie im Menü **Online** auf **Einloggen**.

Durch das Einloggen wird der Online-Modus zum Feldbuscontroller eingeschaltet und die Kommunikationsparameter sind nicht mehr aufrufbar.

Sofern noch kein Programm im Feldbuscontroller vorhanden ist, erscheint ein Fenster mit der Abfrage, ob das Programm geladen werden soll.

8. Um das aktuelle Programm zu laden, bestätigen Sie mit **Ja**.

9. Klicken Sie im Menü **Online** auf **Bootprojekt erzeugen**.

Auf diese Weise wird Ihr kompiliertes Projekt auch ausgeführt, wenn Sie den Feldbuscontroller neu starten oder wenn es einen Spannungsausfall gibt.

10. Wenn das Programm geladen ist, starten Sie die Programmabarbeitung über das Menü **Online** und den Menüpunkt **Start**.

Dieser Befehl startet die Abarbeitung Ihres Programms in der Steuerung bzw. in der Simulation.

Am rechten Ende der Statusleiste wird „ONLINE“ und „LÄUFT“ angezeigt.

11. Um den Online-Betrieb zu beenden, klicken Sie im Menü **Online** auf den Menüpunkt **Ausloggen**.

10 Im Web-Based-Management-System (WBM) konfigurieren

Für die Konfiguration und Verwaltung des Systems stehen Ihnen ein internes Dateisystem und ein integrierter Webserver zur Verfügung, die als Web-Based-Management-System, kurz WBM, bezeichnet werden.

Auf den intern gespeicherten HTML-Seiten erhalten Sie auslesbare Informationen über die Konfiguration und den Status des Feldbusknotens. Außerdem ändern Sie hier die Konfiguration des Gerätes.
Darüber hinaus können Sie über das implementierte Dateisystem auch selbst erstellte HTML-Seiten hinterlegen.

Hinweis

Nach Änderungen an der Konfiguration immer einen Neustart durchführen!
Damit geänderte Konfigurationseinstellungen wirksam werden, führen Sie nach Ihren Änderungen immer einen Systemneustart durch.

1. Zum Öffnen des WBM starten Sie einen Web-Browser (z. B. Microsoft Internet-Explorer oder Mozilla Firefox).

2. Geben Sie in der Adresszeile die IP-Adresse des Feldbuskopplers/ -controllers ein.

3. Bestätigen Sie mit **[Enter]**.
   Die Startseite des WBM wird aufgebaut.

4. Wählen Sie in der linken Navigationsleiste den Link auf die gewünschte HTML-Seite.
   Es erscheint ein Abfragedialog.

5. Geben Sie im Abfragedialog Ihren Benutzernamen und das Passwort ein (standardmäßig: User = „admin“, Passwort = „wago“ oder User = „user“, Passwort = „user“).
   Die entsprechende HTML-Seite wird aufgebaut.

6. Führen Sie die gewünschten Einstellungen durch.

7. Bestätigen Sie Ihre Änderungen mit der Schaltfläche **[SUBMIT]** oder verwerfen Sie diese mit der Schaltfläche **[UNDO]**.

8. Damit die Einstellungen übernommen werden, führen Sie anschließend einen Neustart durch (WBM-Seite „Security“, Schaltfläche **[Software-Reset]**).

Über die Links der Navigationsleiste erreichen Sie die folgenden WBM-Seiten:

- Information
- Ethernet
- TCP/IP
- Port
- Watchdog
- Clock
- Security
- Modbus
- EtherNet/IP
- PLC Info
- PLC
- Features
- I/O config
- Disk Info

10.1 Information

Auf der WBM-Seite „Information“ erhalten Sie eine Übersicht mit allen wichtigen Informationen zu Ihrem Feldbuskoppler/-controller.

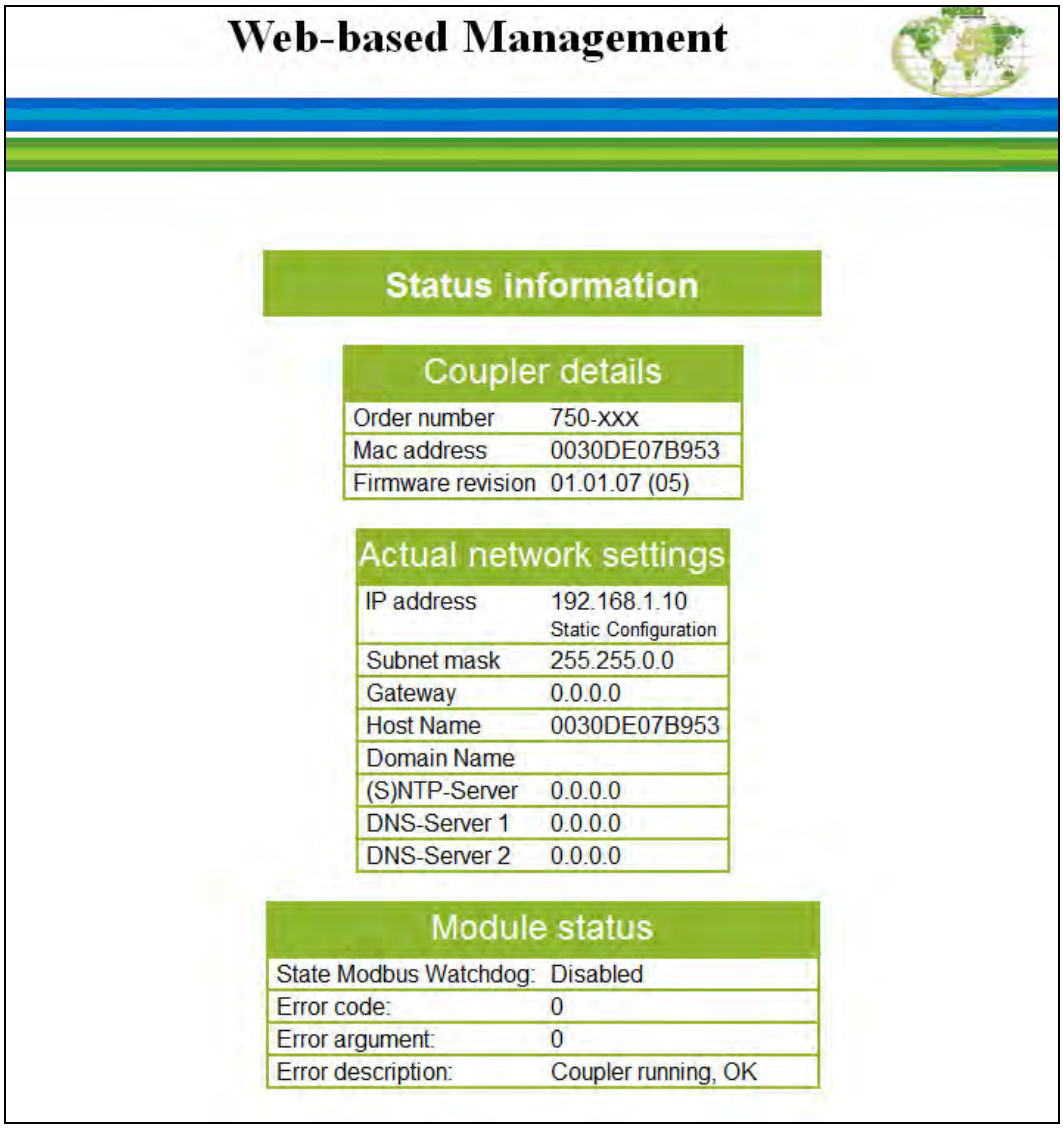


Abbildung 54: WBM-Seite „Information“

Tabelle 36: WBM-Seite „Information“

Coupler details			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
Order number	750-852	750-852	Bestellnummer
Mac address	0030DEXXXXXX	0030DE000006	Hardware MAC-Adresse
Firmware revision	kk.ff.bb (rr)	01.01.09 (00)	Firmware-Revisionsnummer (kk = Kompatibilität, ff = Funktionalität, bb = Bugfix, rr = Revision)
Actual network settings			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
IP address	0.0.0.0	192.168.1.80	IP-Adresse
Subnet mask	255.255.255.0	255.255.255.240	Subnetzmaske
Gateway	0.0.0.0	192.168.1.251	Gateway
Host Name	___	___	Hostname (hier nicht vergeben)
Domain Name	___	___	Domainname (hier nicht vergeben)
(S)NTP-Server	0.0.0.0	0.0.0.0	Adresse (S)NTP-Server
DNS-Server 1	0.0.0.0	0.0.0.0	Adresse erster DNS-Server
DNS-Server 2	0.0.0.0	0.0.0.0	Adresse zweiter DNS-Server
Module status			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
State Modbus Watchdog	Disabled	Disabled	Status des Modbus Watchdog
Error code	0	10	Fehlercode
Error argument	0	5	Fehlerargument
Error description	Coupler running, OK	Mismatch in CODESYS IO-configuration	Fehlerbeschreibung

10.2 Ethernet

Auf der Seite „Ethernet“ konfigurieren Sie die Übertragungsrate und die Bandbreitenbegrenzung für die ETHERNET-Kommunikation.

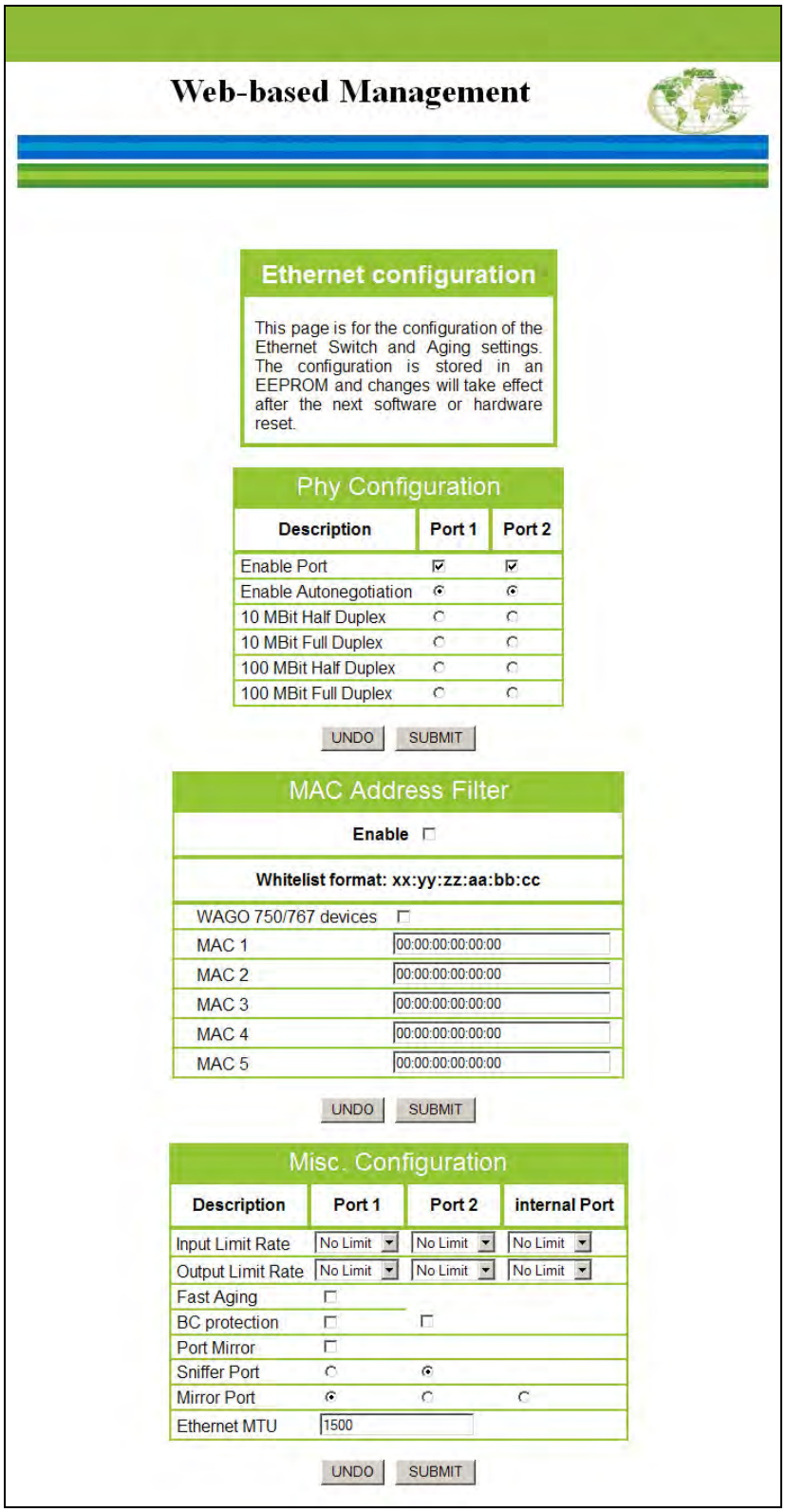


Abbildung 55: WBM-Seite „Ethernet“

Tabelle 37: WBM-Seite „Ethernet“

Phy Configuration			
Eintrag	**Standard-wert**	**Beschreibung**	
Enable Port	☑	☑	Port 1, Port 2 aktivieren
		☐	Port 1, Port 2 deaktivieren
Enable Autonegotiation	⊙	⊙	Autonegotiation aktivieren Der ETHERNET-Übertragungsmodus wird automatisch an den Kommunikationspartner angepasst.
		○	Autonegotiation deaktivieren
10 MBit Half Duplex 10 MBit Full Duplex 100 MBit Half Duplex 100 MBit Full Duplex	○ ○ ○ ○	Verwenden eines festen ETHERNET-Übertragungsmodus mit 10/100 MBit Halb-/Vollduplex.	

MAC Address Filter			
Eintrag	**Standard-wert**	**Beschreibung**	
Always allow WAGO devices	☐	☑	MAC-Adressenfilter aktivieren. Nur die nachfolgend eingetragenen MAC-Adressen haben Netzwerkzugriff auf den Feldbuscontroller, alle anderen werden geblockt.
		☐	MAC-Adressenfilter deaktivieren. Alle MAC-Adressen haben Netzwerkzugriff auf den Feldbuscontroller.
MAC 1	00:00:00:00:00:00	Filter für die erste MAC-Adresse (hexadezimal).	
MAC 2	00:00:00:00:00:00	Filter für die zweite MAC-Adresse (hexadezimal).	
MAC 3	00:00:00:00:00:00	Filter für die dritte MAC-Adresse (hexadezimal).	
MAC 4	00:00:00:00:00:00	Filter für die vierte MAC-Adresse (hexadezimal).	
MAC 5	00:00:00:00:00:00	Filter für die fünfte MAC-Adresse (hexadezimal).	

Misc. Configuration					
Eintrag	**Port**			**Beschreibung**	
	1	**2**	**internal**		
Input Limit Rate	No Limit ▼			Die Input-Limit-Rate begrenzt den Netzwerkverkehr beim Empfangen. Die Rate wird in Megabit pro Sekunde bzw. Kilobit pro Sekunde angegeben. Wird die Begrenzung überschritten, werden Pakete verworfen.	
Output Limit Rate	No Limit ▼			Die Output-Limit-Rate begrenzt den Netzwerkverkehr beim Senden. Die Rate wird in Megabit pro Sekunde bzw. Kilobit pro Sekunde angegeben. Wird die Begrenzung überschritten, werden Pakete verworfen.	
Fast Aging	☐			☑	Fast Aging aktivieren. Fast Aging sorgt dafür, dass der Cache für die MAC-Adressen im Switch schneller verworfen wird. Dieses kann erforderlich sein, wenn ein Redundanzsystem (z. B. mittels Jet-Ring-Netzwerk oder vergleichbarer Technik) aufgebaut werden soll.
				☐	Fast Aging deaktivieren. Der Zeitraum bis zum Verwerfen der Cache-Einträge beträgt fünf Minuten.

BC protection	☐	☐		☑ Die Broadcast-Protection begrenzt die Anzahl der Broadcast-Telegramme pro Zeiteinheit. Wird die Protection eingeschaltet, werden die Broadcast-Pakete bei 100 MBit auf 8 Pakete pro 10 ms begrenzt und bei 10 MBit auf 8 Pakete pro 100 ms. Wird die Begrenzung überschritten, werden Pakete verworfen. ☐ Broadcast-Protection nicht aktiv.
Port Mirror	☐			☑ Port-Mirroring aktivieren Das Port-Mirroring dient zur Netzwerkdiagnose. Hier werden Pakete von einem Port (Mirror Port) auf einen anderen Port (Sniffer Port) gespiegelt. ☐ Port-Mirroring deaktivieren
Sniffer Port	○	⊙		Auswahl des Sniffer-Ports, auf den der Mirror-Port gespiegelt werden soll.
Mirror Port	⊙	○	○	Auswahl des Mirror-Ports, der auf den Sniffer-Port gespiegelt werden soll.
Ethernet MTU	1500			Maximale Paketgröße eines Protokolls, welche ohne Fragmentierung übertragen werden kann („Maximum Transmission Unit“ – MTU)

Hinweis

MTU-Wert nur für Fragmentierung einstellen!
Stellen Sie nur dann den Wert für MTU, d. h. die maximale, zwischen Client und Server vereinbarte Paketgröße, entsprechend ein, wenn Sie ein Tunnel-Protokoll (z. B. VPN) für die ETHERNET-Kommunikation verwenden und die Pakete fragmentiert werden müssen. Diese Wert-Einstellung ist unabhängig von dem gewählten Übertragungsmodus.

Hinweis

ETHERNET-Übertragungsmodus korrekt konfigurieren!
Eine fehlerhafte Konfiguration des ETHERNET-Übertragungsmodus kann einen Verbindungsverlust, eine schlechte Netzwerk-Performance oder ein fehlerhaftes Verhalten des Feldbuskopplers/-controllers zur Folge haben.

Hinweis

Nicht alle ETHERNET-Ports deaktivierbar!
Beachten Sie, dass es nicht möglich ist, beide ETHERNET-Ports abzuschalten. Sind beide Ports deaktiviert und Sie klicken auf die Schaltfläche **[SUBMIT]**, wird die Auswahl nicht angenommen und die vorherigen Werte werden wieder hergestellt.

10.3 TCP/IP

Auf der HTML-Seite „TCP/IP“ konfigurieren Sie die Netzwerkadressierung und die Netzwerkidentifikation.

Hinweis

DIP-Schalter auf „0“ und „use IP from EEPROM“ aktiv schalten!
Bevor Sie auf dieser Seite Parameter ändern, stellen Sie den DIP-Schalter auf Null, und aktivieren Sie auf der WBM-Seite „Port“ die Option: „use IP from EEPROM“!
Sind diese Voraussetzungen nicht gegeben, wird stattdessen die Einstellung des DIP-Schalters übernommen.

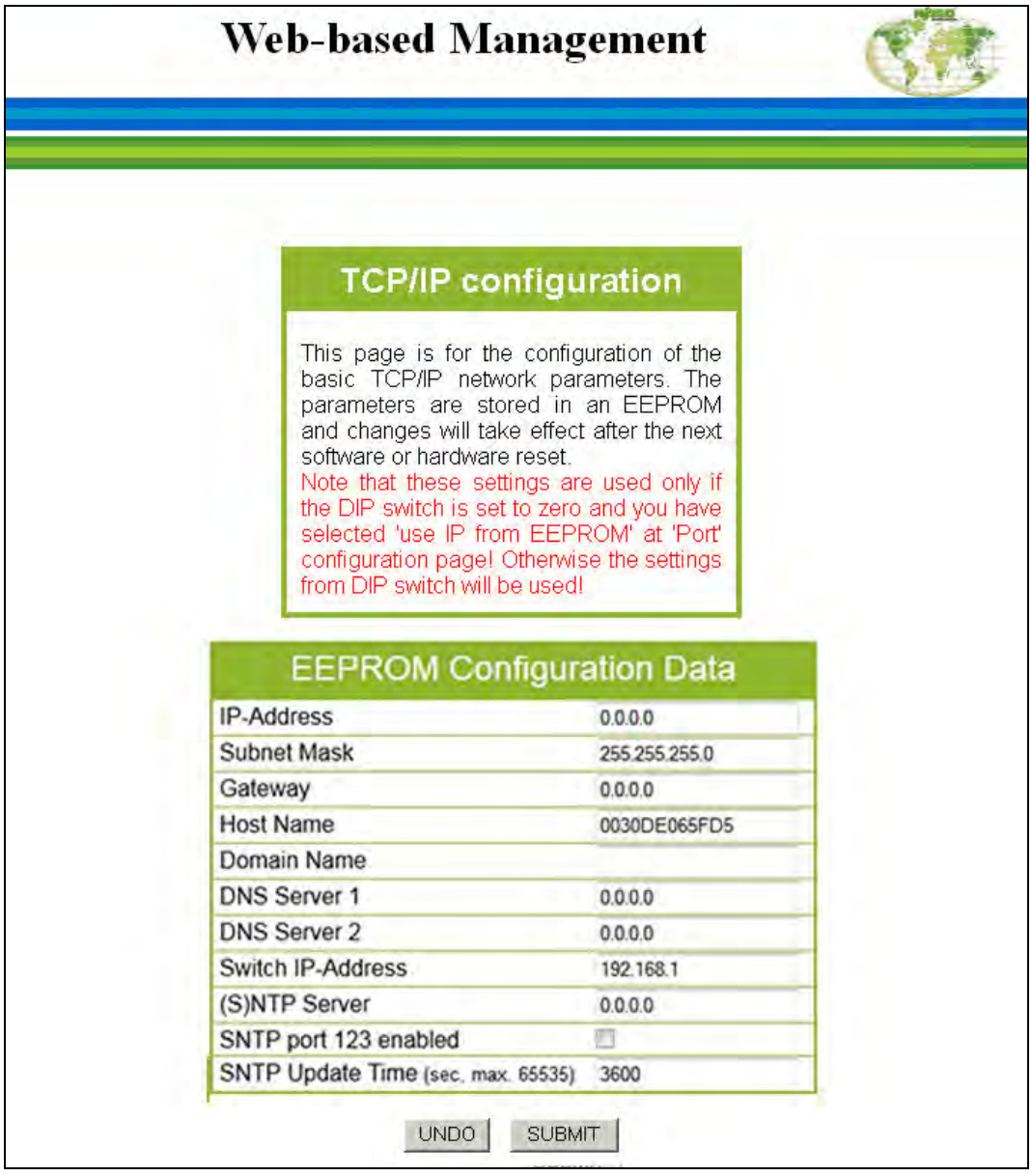

Web-based Management
TCP/IP configuration
This page is for the configuration of the basic TCP/IP network parameters. The parameters are stored in an EEPROM and changes will take effect after the next software or hardware reset.
Note that these settings are used only if the DIP switch is set to zero and you have selected 'use IP from EEPROM' at 'Port' configuration page! Otherwise the settings from DIP switch will be used!
EEPROM Configuration Data
IP-Address 0.0.0.0
Subnet Mask 255.255.255.0
Gateway 0.0.0.0
Host Name 0030DE065FD5
Domain Name
DNS Server 1 0.0.0.0
DNS Server 2 0.0.0.0
Switch IP-Address 192.168.1
(S)NTP Server 0.0.0.0
SNTP port 123 enabled
SNTP Update Time (sec. max. 65535) 3600
UNDO
SUBMIT

Abbildung 56: WBM-Seite „TCP/IP“

Tabelle 38: WBM-Seite „TCP/IP“

EEPROM Configuration Data			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
IP-Address	0.0.0.0	192.168.1.180	IP-Adresse eintragen
Subnet Mask	255.255.255.0	255.255.255.0	Subnetzmaske eintragen
Gateway	0.0.0.0	0.0.0.0	Gateway eintragen
Host Name	0030DEXXX XXX	0030DE026005	Hostname eintragen
Domain Name			Domainname eintragen
DNS-Server 1	0.0.0.0	0.0.0.0	IP-Adresse des ersten DNS-Servers eintragen
DNS-Server 2	0.0.0.0	0.0.0.0	Optionale IP-Adresse des zweiten DNS-Servers eintragen
Switch IP-Address	192.168.1	192.168.5	Netzadresse für die Konfiguration der IP-Adresse über den DIP-Schalter
(S)NTP-Server	0.0.0.0	0.0.0.0	IP-Adresse des (S)NTP-Servers eintragen
SNTP port 123 enabled	☐	☑	„Simple Network Time Protocol“ aktivieren
		☐	„Simple Network Time Protocol“ deaktivieren
SNTP Update Time (sec. max. 65535)	3600	0	Wartezeit zwischen zwei Anfragen des SNTP-Clients, nach welcher die Netzwerkzeit neu vom (S)NTP-Server angefordert wird, eintragen

10.4 Port

Auf der HTML-Seite „Port“ aktivieren oder deaktivieren Sie die über das IP-Protokoll verfügbaren Dienste.

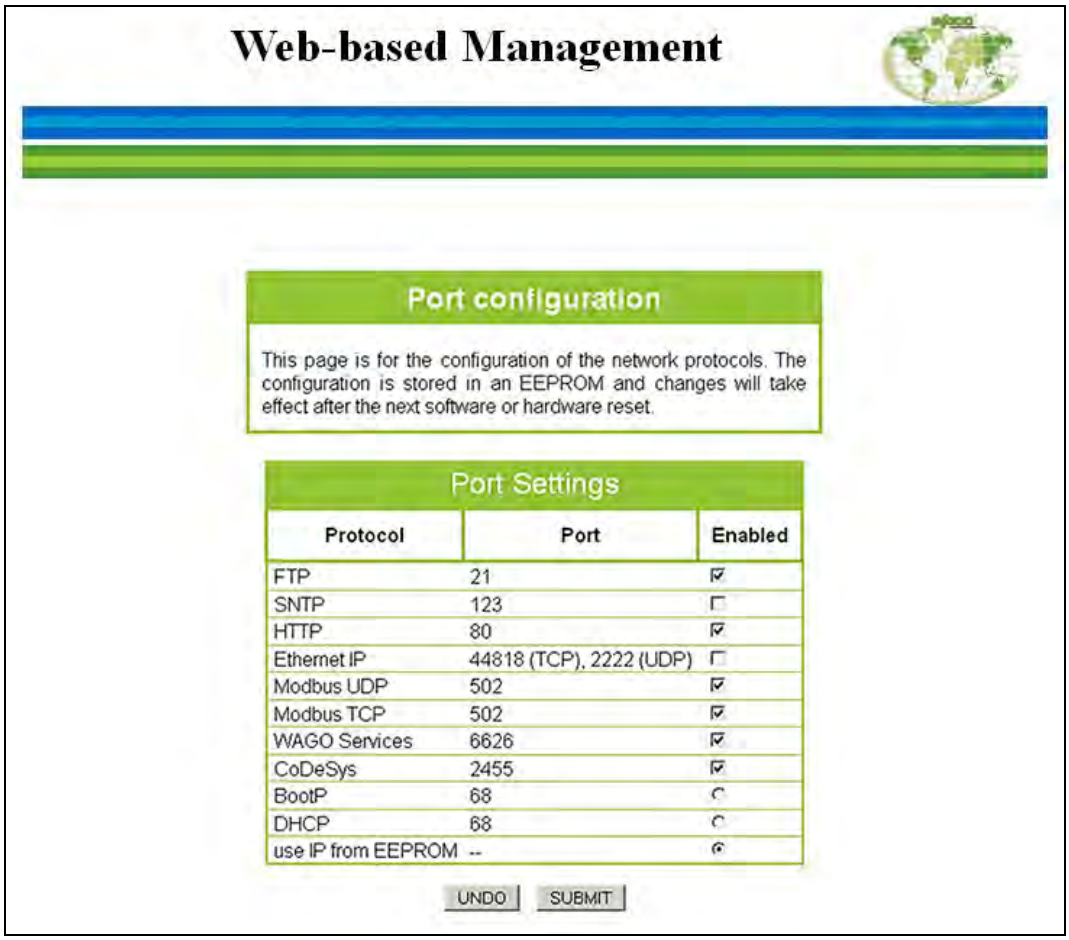


Abbildung 57: WBM-Seite „Port“

Tabelle 39: WBM-Seite „Port“

Port Settings		
Eintrag	**Standardwert**	**Beschreibung**
FTP (Port 21)	Enabled ☑	☑ „File Transfer Protocol“ aktivieren
		☐ „File Transfer Protocol“ deaktivieren
SNTP (Port 123)	Enabled ☐	☑ „Simple Network Time Protocol“ aktivieren
		☐ „Simple Network Time Protocol“ deaktivieren
HTTP (Port 80)	Enabled ☑	☑ „Hypertext Transfer Protocol“ aktivieren
		☐ „Hypertext Transfer Protocol“ deaktivieren
Ethernet IP (TCP-Port 44818, UDP-Port 2222)	Enabled ☐	☑ ETHERNET/IP-Protokoll aktivieren
		☐ ETHERNET/IP-Protokoll deaktivieren
Modbus UDP (Port 502)	Enabled ☑	☑ MODBUS/UDP-Protokoll aktivieren
		☐ MODBUS/UDP-Protokoll deaktivieren
Modbus TCP (Port 502)	Enabled ☑	☑ MODBUS/TCP-Protokoll aktivieren
		☐ MODBUS/TCP-Protokoll deaktivieren
WAGO Services (Port 6626)	Enabled ☑	☑ WAGO Services aktivieren
		☐ WAGO Services deaktivieren
CODESYS (Port 2455)	Enabled ☑	☑ CODESYS aktivieren
		☐ CODESYS deaktivieren
BootP (Port 68)	Enabled ◉	◉ „Bootstrap Protocol“ aktivieren
		○ „Bootstrap Protocol“ deaktivieren
DHCP (Port 68)	Enabled ○	◉ „Dynamic Host Configuration Protocol“ aktivieren
		○ „Dynamic Host Configuration Protocol“ deaktivieren
use IP from EEPROM	Enabled ○	◉ IP-Adrese aus dem EEPROM verwenden
		○ IP-Adrese nicht aus dem EEPROM verwenden

Hinweis

Alternative IP-Adressvorgabe!
Sie können die Einstellung „DHCP“, „BootP“ und „use IP from EEPROM“ nur alternativ auswählen!

10.5 Watchdog

Auf der HTML-Seite „Watchdog“ nehmen Sie Einstellungen für den Connection- und MODBUS-Watchdog vor.

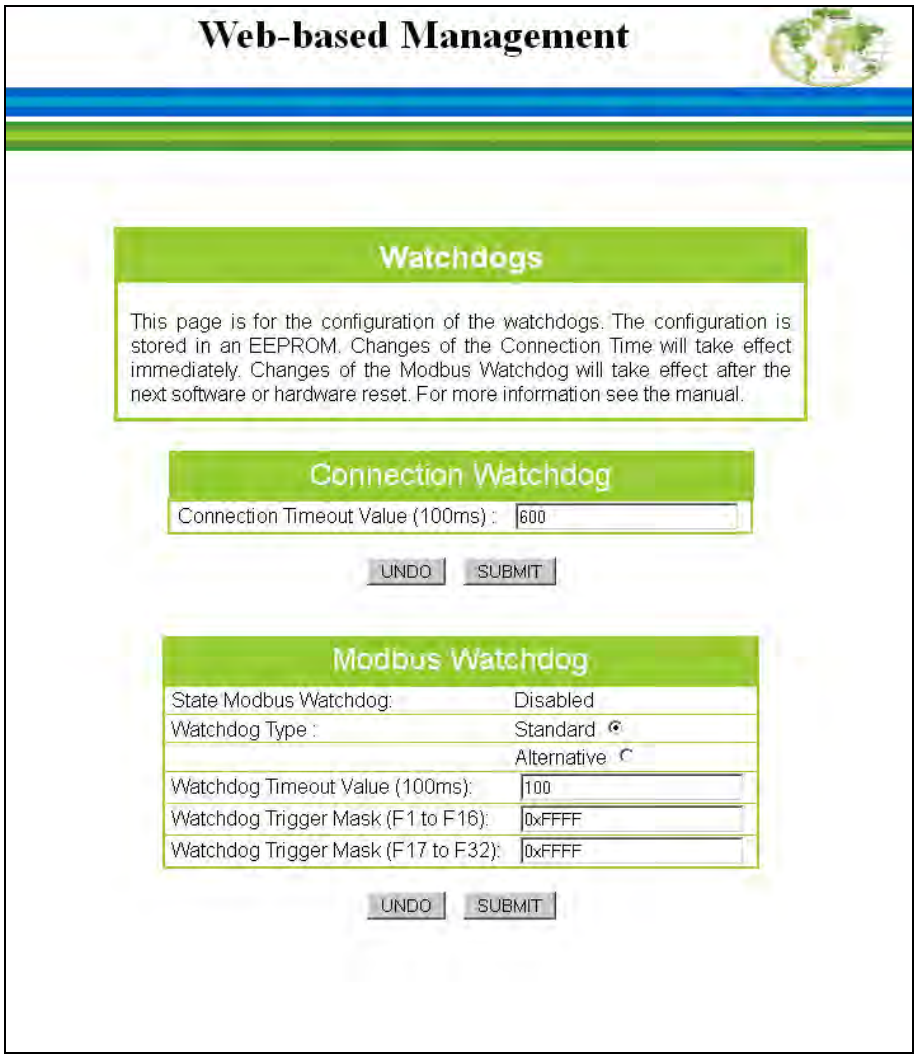

Web-based Management
Watchdogs
This page is for the configuration of the watchdogs. The configuration is stored in an EEPROM. Changes of the Connection Time will take effect immediately. Changes of the Modbus Watchdog will take effect after the next software or hardware reset. For more information see the manual.
Connection Watchdog
Connection Timeout Value (100ms) :
600
UNDO
SUBMIT
Modbus Watchdog
State Modbus Watchdog:
Disabled
Watchdog Type :
Standard
Alternative
Watchdog Timeout Value (100ms):
100
Watchdog Trigger Mask (F1 to F16):
0xFFFF
Watchdog Trigger Mask (F17 to F32):
0xFFFF
UNDO
SUBMIT

Abbildung 58: WBM-Seite „Watchdog“

Tabelle 40: WBM-Seite „Watchdog“

Connection Watchdog		
Eintrag	**Standardwert**	**Beschreibung**
Connection Timeout Value (100 ms)	600	Überwachungszeit für TCP-Verbindungen. Nach Ablauf dieser Zeit ohne erfolgten Datenverkehr wird die TCP-Verbindung geschlossen.
Modbus Watchdog		
Eintrag	**Standardwert**	**Beschreibung**
State Modbus Watchdog	Disabled	„Enabled“ – Watchdog aktiviert „Disabled“ – Watchdog deaktiviert
Watchdog Type	Standard ◉	Die eingestellte Codiermaske (Watchdog Trigger Mask) wird ausgewertet, um zu entscheiden, ob die Watchdog-Zeit zurückzusetzen ist.
	Alternative ○	Mit jedem beliebigen MODBUS/TCP-Telegramm wird die Watchdog-Zeit zurückgesetzt.
Watchdog Timeout Value (100 ms)	100	Überwachungszeit für MODBUS-Verbindungen. Nach Ablauf dieser Zeit ohne empfangenes MODBUS-Telegramm, werden die physikalischen Ausgänge auf '0' gesetzt.
Watchdog Trigger Mask (F1 to F16)	0xFFFF	Codiermaske für bestimmte MODBUS-Telegramme (Function Code FC1 ... FC16)
Watchdog Trigger Mask (F17 to F32)	0xFFFF	Codiermaske für bestimmte MODBUS-Telegramme (Function Code FC17 ... FC32)

10.6 Clock

Auf der HTML-Seite „Clock“ nehmen Sie Einstellungen für die Feldbuscontrollerinterne Systemzeit vor. Geben Sie hier die aktuelle Uhrzeit und das Datum ein und wählen Sie Winter- oder Sommerzeit aus.

Hinweis

Systemzeit wird im spannungslosen Zustand nicht weitergeführt!
Der Feldbuscontroller 750-852 verfügt über keinen Echtzeit-Uhrenbaustein. Aus diesem Grund wird die aktuelle Systemzeit im spannungslosen Zustand nicht weitergeführt!

Nach dem Einschalten der Betriebsspannung beginnt die Systemzeit bei 01.01.2000 00:00:00 Uhr.

Synchronisieren Sie die Systemzeit nach dem Einschalten der Betriebsspannung mit der aktuellen Rechnerzeit.

Hinweis

WAGO-RTC-Klemme zur Zeitsynchronisierung nutzen!
Sie können eine WAGO-RTC-Klemme 750-640 in Ihrem Knoten verwenden, um die aktuelle Zeit (Realtime Clock – RTC) in codierter Form in Ihrer übergeordneten Steuerung zu nutzen.

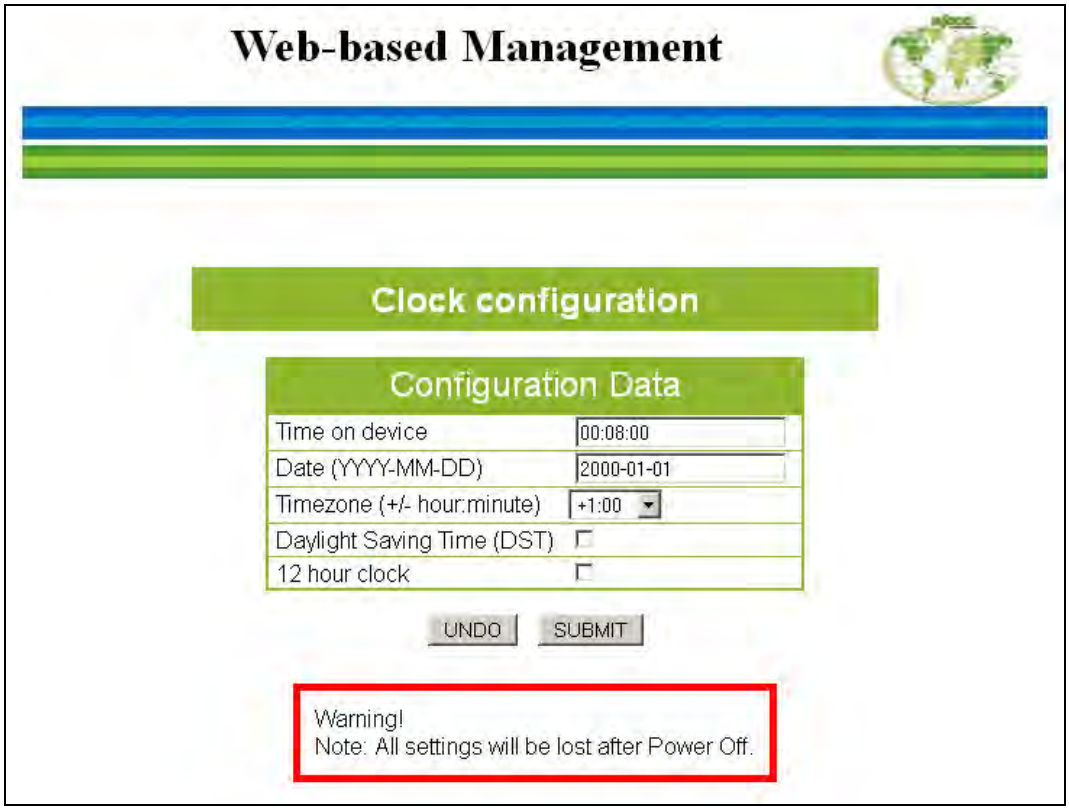


Abbildung 59: WBM-Seite „Clock“

Tabelle 41: WBM-Seite „Clock“

Configuration Data			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
Time on device	Koordinierte Weltzeit UTC	13:00:31	aktuelle Uhrzeit einstellen
Date (YYYY-MM-DD)	Datum abhängig von der UTC	2013-02-08	aktuelles Datum einstellen
Timezone (+/- hour:minute)	0:00	1 (MEZ)	Zeitzonenabweichung von der koordinierten Weltzeit (UTC) einstellen
Daylight Saving Time (DST)	☐	☑	☑ Sommerzeit aktivieren ☐ Winterzeit aktivieren
12 hour clock	☐	☑	☑ 12-Stunden-Anzeige aktivieren ☐ 24-Stunden- Anzeige aktivieren

10.7 Security

Auf der HTML-Seite „Security“ richten Sie durch Passwörter Lese- und/oder Schreibzugriffe für verschiedene Anwendergruppen zum Schutz vor Konfigurationsänderungen ein.

Hinweis

Passwortänderung nur durch „admin“ und nach Software-Reset möglich!
Sie können nur über den Benutzer „admin“ und dem zugehörigen Passwort die Passwörter ändern.
Damit die geänderten Einstellungen wirksam werden, führen Sie mit der Schaltfläche **[Software Reset]** einen Software-Neustart durch.

Hinweis

Passwort-Restriktionen beachten!
Für Passwörter gilt folgende Einschränkung:
- max. 32 Zeichen mit Sonderzeichen.

Hinweis

Nach Software-Reset Zugriff erneuern!
Wenn Sie auf dieser Seite einen Software-Reset auslösen, dann startet der Feldbuskoppler/-controller mit den Konfigurationen, die zuvor ins EEPROM geladen wurden, und die Verbindung zum Browser wird unterbrochen.
Haben Sie zuvor die IP-Adresse geändert, müssen Sie mit der geänderten IP-Adresse über den Browser auf das Gerät zugreifen.
Haben Sie die IP-Adresse nicht geändert, sondern andere Einstellungen durchgeführt, können Sie durch Aktualisieren des Browsers die Verbindung wieder herstellen.

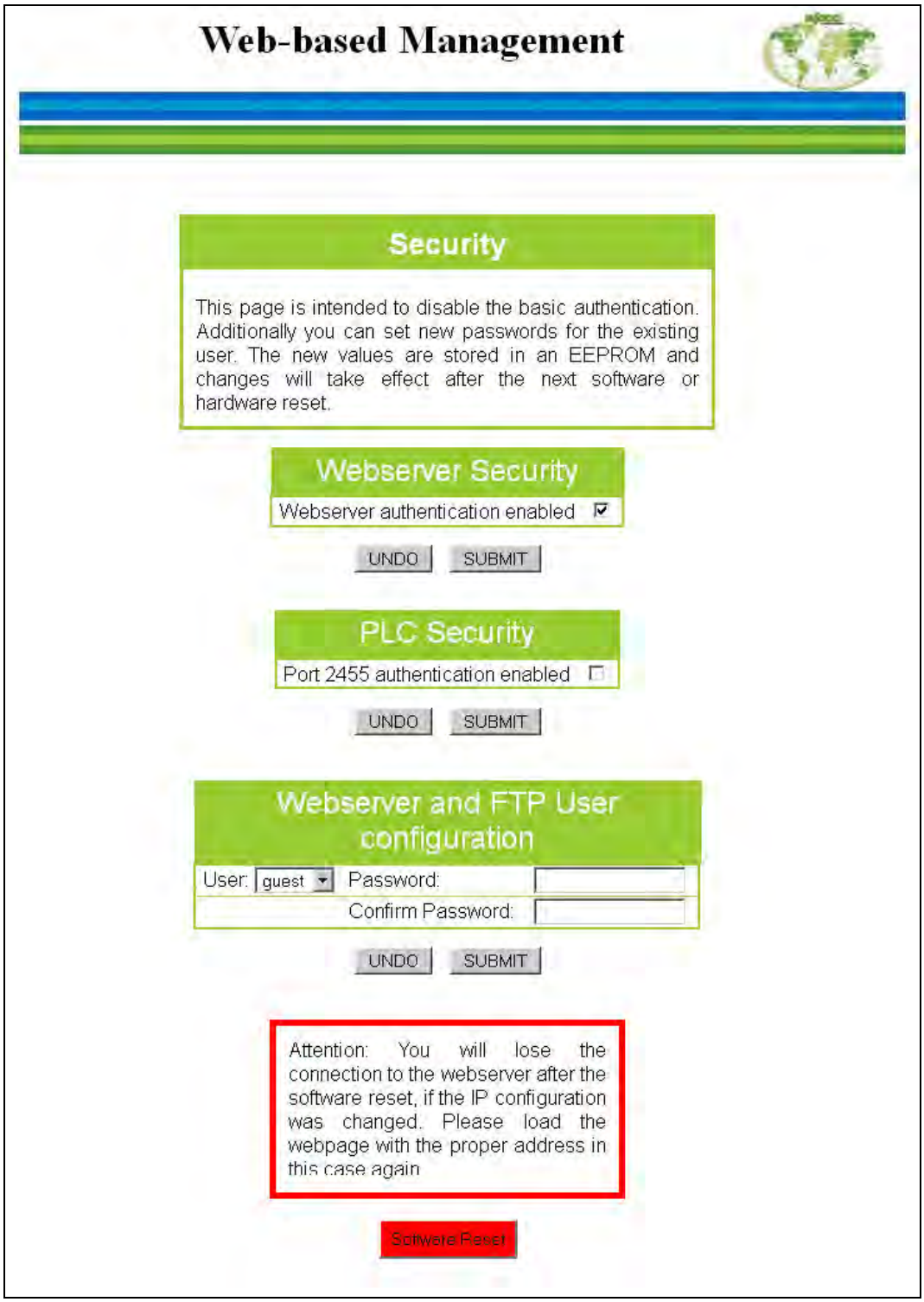

Web-based Management
Security
This page is intended to disable the basic authentication. Additionally you can set new passwords for the existing user. The new values are stored in an EEPROM and changes will take effect after the next software or hardware reset.
Webserver Security
Webserver authentication enabled
UNDO
SUBMIT
PLC Security
Port 2455 authentication enabled
UNDO
SUBMIT
Webserver and FTP User configuration
User: guest
Password:
Confirm Password:
UNDO
SUBMIT
Attention: You will lose the connection to the webserver after the software reset, if the IP configuration was changed. Please load the webpage with the proper address in this case again
Software Reset

Abbildung 60: WBM-Seite „Security“

Tabelle 42: WBM-Seite „Security“

Webserver Security		
Eintrag	**Standardwert**	**Beschreibung**
Webserver authentication enabled	☑	☑ Passwortschutz für den Zugriff auf das Web-Interface aktivieren
		☐ Passwortschutz für den Zugriff auf das Web-Interface deaktivieren

Webserver, FTP and PLC User configuration *)		
Eintrag	**Standardwert**	**Beschreibung**
User	*)	admin, guest oder user auswählen
Password	*)	Passwort eintragen
Confirm Password		Passwort erneut zur Bestätigung eintragen

PLC Security		
Eintrag	**Standardwert**	**Beschreibung**
Port 2455 authentication enabled	☐	☑ Passwortschutz für den Zugriff auf das Anwendungsprogramm aktivieren
		☐ Passwortschutz für den Zugriff auf das Anwendungsprogramm deaktivieren

***)** Standardmäßig sind folgende Gruppen vorgesehen:

User: admin	Passwort: wago
User: guest	Passwort: guest
User: user	Passwort: user

10.8 Modbus

Auf der HTML-Seite „Modbus“ nehmen Sie Einstellungen für das MODBUS-Protokoll vor.

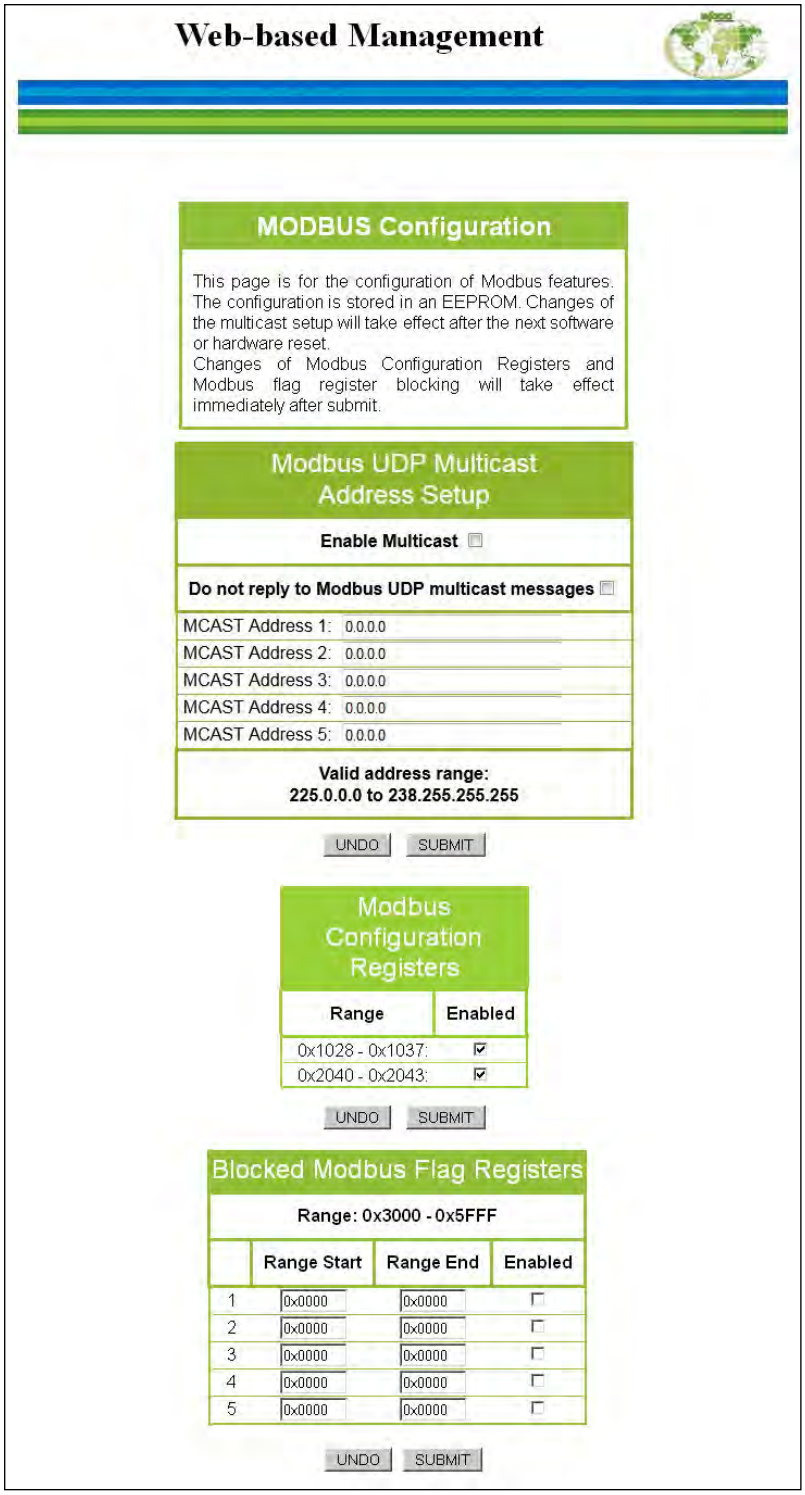


Abbildung 61: WBM-Seite „MODBUS“

Tabelle 43: WBM-Seite „Modbus“

Modbus UDP Multicast Address Setup		
Eintrag	**Standardwert**	**Beschreibung**
Enable Multicast	☐	☑ Multicast für die MODBUS UDP-Übertragung aktivieren. Zusätzlich zur eigenen IP-Adresse nimmt der Feldbuscontroller auch für die nachfolgend eingetragenen MCAST-Adressen Modbus-Kommandos entgegen.
		☐ Kein Multicast für die MODBUS UDP-Übertragung aktiviert. Der Feldbuscontroller nimmt nur für die eigene IP-Adresse Modbus-Kommandos entgegen.
Do not reply to Modbus UDP multicast messages	☐	☑ Das Beantworten von Modbus UDP multicast-Nachrichten ist deaktiviert.
		☐ Das Beantworten von Modbus UDP multicast-Nachrichten ist aktiviert.
MCAST Address 1 ... 5:	0.0.0.0	Multicast-Adresse 1... 5, für die Multicast aktiviert sein soll. Der gültige Adressbereich ist im WBM angegeben. Doppelte Adressen sind nicht erlaubt.
Modbus Configuration Registers		
Eintrag	**Standardwert**	**Beschreibung**
Range 0x1028 – 0x1037 0x2040 – 0x2043	☑ ☑	☑ MODBUS-Konfigurationsregisterbereich aktivieren.
		☐ MODBUS-Konfigurationsregisterbereich deaktiviert.
Blocked Modbus Flag Registers		
Eintrag 1 … 5	**Standardwert**	**Beschreibung**
Range Start	0x0000	ModbusTCP-Startadresse für den Sperrbereich.
Range End	0x0000	ModbusTCP-Endadresse für den Sperrbereich.
Enabled	☐	☑ Anfragen auf den jeweils eingetragenen Adressbereich des ModbusTCP-Merkerbereiches werden mit dem Exceptioncode 0x02 „Illegal Data Address“ beantwortet. Liegt eine Anfrage teilweise im gültigen und teilweise im gesperrten Bereich, wird diese ebenfalls mit einer Exception beantwortet.
		☐ Der jeweilige Tabelleneintrag wird nicht berücksichtigt.

Hinweis

Multicast-Funktion nur mit gültiger MCAST-Adresse aktiv!
Aktivieren Sie die Funktion „**Enable Multicast**“, ohne dass Sie eine Adresse ungleich 0.0.0.0 eintragen, wird die Funktion nach Klicken auf die Schaltfläche **[SUBMIT]** automatisch wieder deaktiviert.
Ist die Funktion „**Enable Multicast**“ bereits mit gültigen Adressen aktiviert, können Sie diese MCAST-Adressfelder anschließend nicht mit ungültigen Adressen beschreiben, da ein Klicken auf die Schaltfläche **[SUBMIT]** wieder auf die zuletzt gültigen Adressen zurückstellt. Die Funktion „**Enable Multicast**“ behält dabei ihren Zustand bei.

10.9 EtherNet/IP

Auf der Seite „EtherNet/IP“ konfigurieren Sie das optionale Auffüllen der statischen Assembly Instanzen und erhalten eine Aufstellung mit der vorliegenden Datenverteilung für den Austausch von Prozessdaten über die EtherNet/IP-Kommunikation.

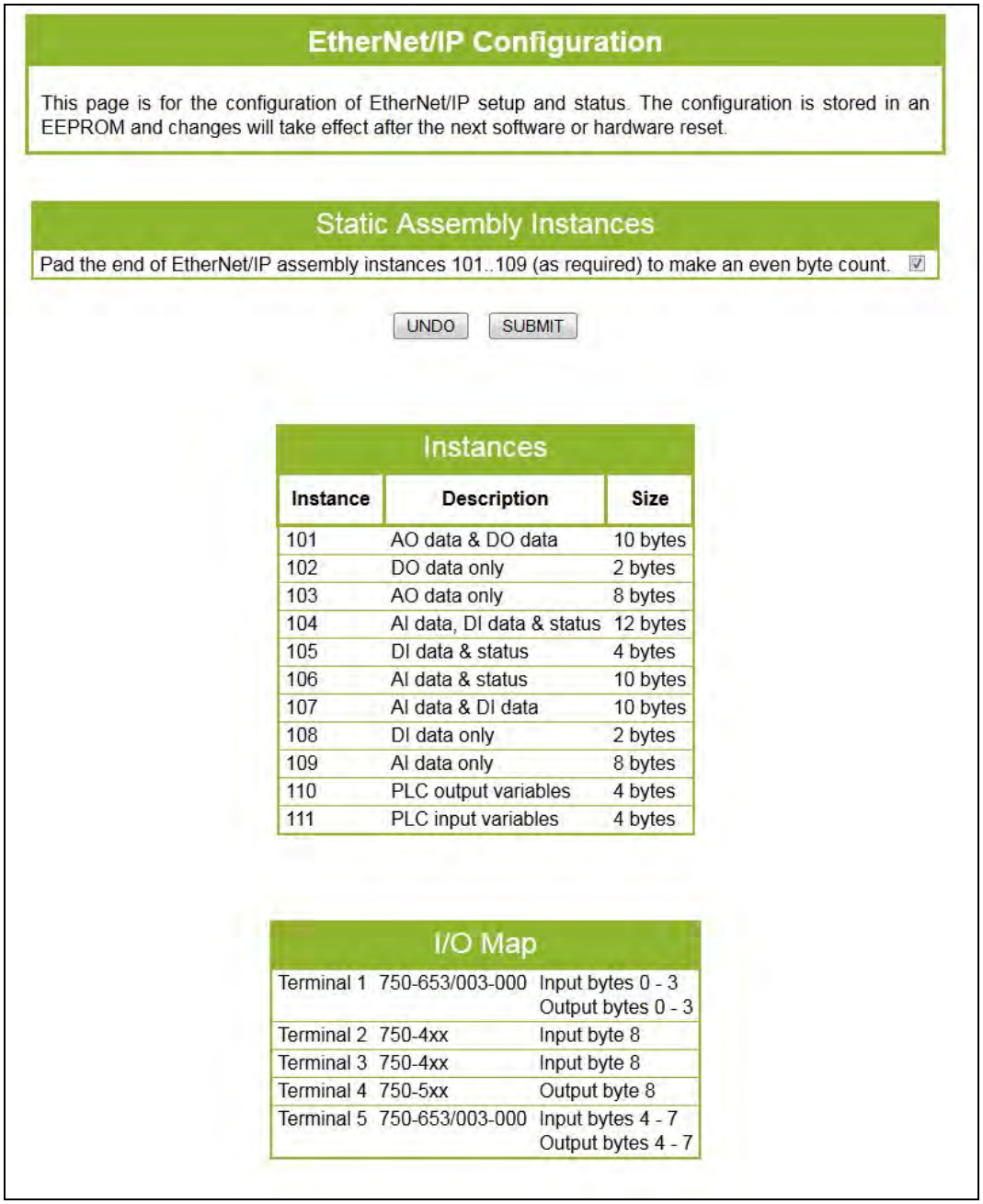

EtherNet/IP Configuration
This page is for the configuration of EtherNet/IP setup and status. The configuration is stored in an EEPROM and changes will take effect after the next software or hardware reset.
Static Assembly Instances
Pad the end of EtherNet/IP assembly instances 101..109 (as required) to make an even byte count.
UNDO
SUBMIT
Instances
Instance | Description | Size
101 | AO data & DO data | 10 bytes
102 | DO data only | 2 bytes
103 | AO data only | 8 bytes
104 | AI data, DI data & status | 12 bytes
105 | DI data & status | 4 bytes
106 | AI data & status | 10 bytes
107 | AI data & DI data | 10 bytes
108 | DI data only | 2 bytes
109 | AI data only | 8 bytes
110 | PLC output variables | 4 bytes
111 | PLC input variables | 4 bytes
I/O Map
Terminal 1 | 750-653/003-000 | Input bytes 0 - 3, Output bytes 0 - 3
Terminal 2 | 750-4xx | Input byte 8
Terminal 3 | 750-4xx | Input byte 8
Terminal 4 | 750-5xx | Output byte 8
Terminal 5 | 750-653/003-000 | Input bytes 4 - 7, Output bytes 4 - 7

Abbildung 62: WBM-Seite „EtherNet/IP“

Tabelle 44: WBM-Seite „EtherNet/IP“

Static Assembly Instances			
Eintrag	**Standardwert**	**Beschreibung**	
Pad the end of EtherNet/IP assembly instances 101…109 (as required) to make an even byte count.	☐	☑ Das Auffüllen der EtherNet/IP-Assemby-Instanzen 101 … 109 ist aktiviert, um eine gerade Byte-Anzahl zu erreichen.	
		☐ Kein Auffüllen der Ethernet/IP-Assembly-Instanzen.	
Instances			
Instance/Description		**Beschreibung**	**Datengröße**
101 … 109	AO data & DO data … AI data only	Analog- und Digitalausgangsdaten … nur Analogeingangsdaten	10 bytes … 8 bytes
110	PLC output variables	SPS-Ausgangsvariablen	4 bytes
111	PLC input variables	SPS-Eingangsvariablen	4 bytes
I/O Map			
Eintrag		**Beispiel gesteckte Busklemmen**	**Datentyp und -größe**
Terminal 1		750-653/003-000	Input bytes 0 - 3 Output bytes 0 - 3
Terminal 2		750-4xx	Input byte 8
…			

10.10 PLC-Info

Auf der HTML-Seite „PLC-Info“ erhalten Sie Informationen zu dem aktuellen CODESYS-Projekt. Voraussetzung für die Anzeige der Informationen ist, dass diese zuvor in CODESYS unter dem Menü „Projekt“→ Menüpunkt „Projekt-informationen“ eingetragen wurden.

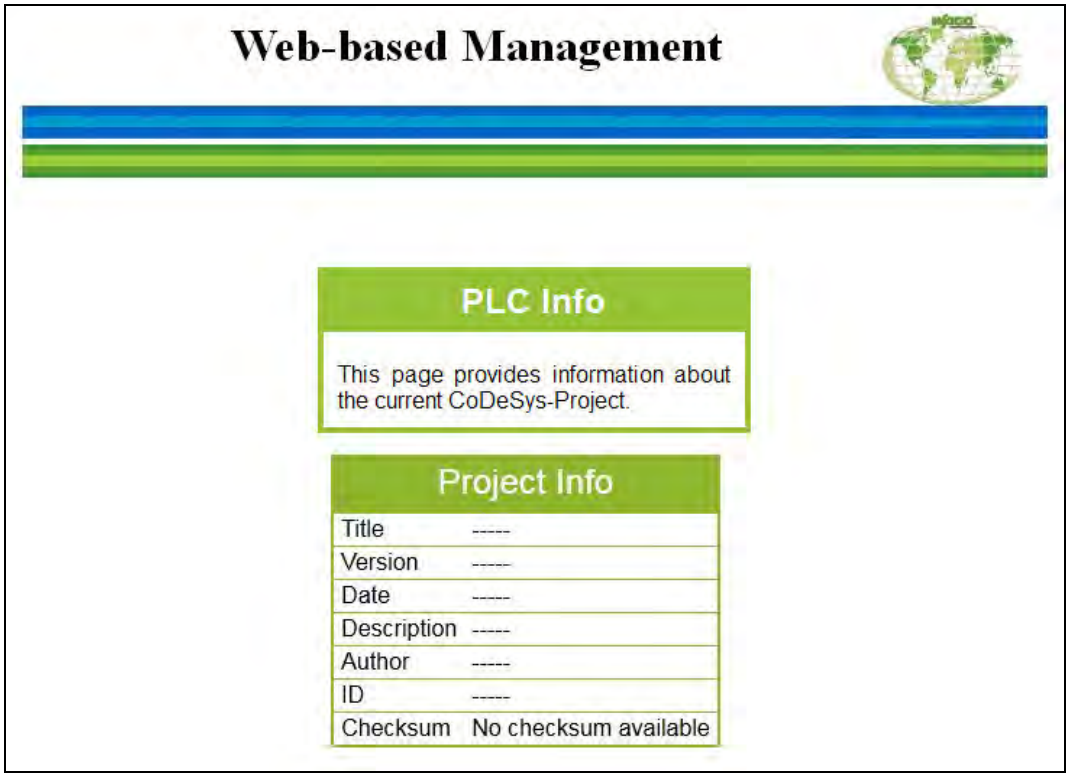


Abbildung 63: WBM-Seite „PLC-Info“

Tabelle 45: WBM-Seite „PLC-Info“

Project-Info			
Eintrag	**Standardwert**	**Wert (Beispiel)**	**Beschreibung**
Title	____	SSL Client Example	Projekttitel
Version	____	1.0.0	Projektversion
Date	____	15.05.2012 08:50:27	Datum und Uhrzeit des Projektes
Description	____	Testclient zum Aufbau von SSL-Verbindungen	Beschreibung
Author	____	JW	Projektersteller
ID	____	70632	Projekt-ID
Checksum	No checksum available		Checksumme des Boot-Projektes auf dem Feldbuscontroller. Die Checksumme wird aus den ausgelesenen Daten als unsigned long 32-Bit-Zahl im Little-Endian-Format gebildet und dezimal angezeigt. Ist die Datei „default.chk“ nicht verfügbar, wird „No checksum available“ angezeigt.

10.11 PLC

Auf der HTML-Seite „PLC“ nehmen Sie Einstellungen für die PFC-Funktionalität Ihres Feldbuscontrollers vor.

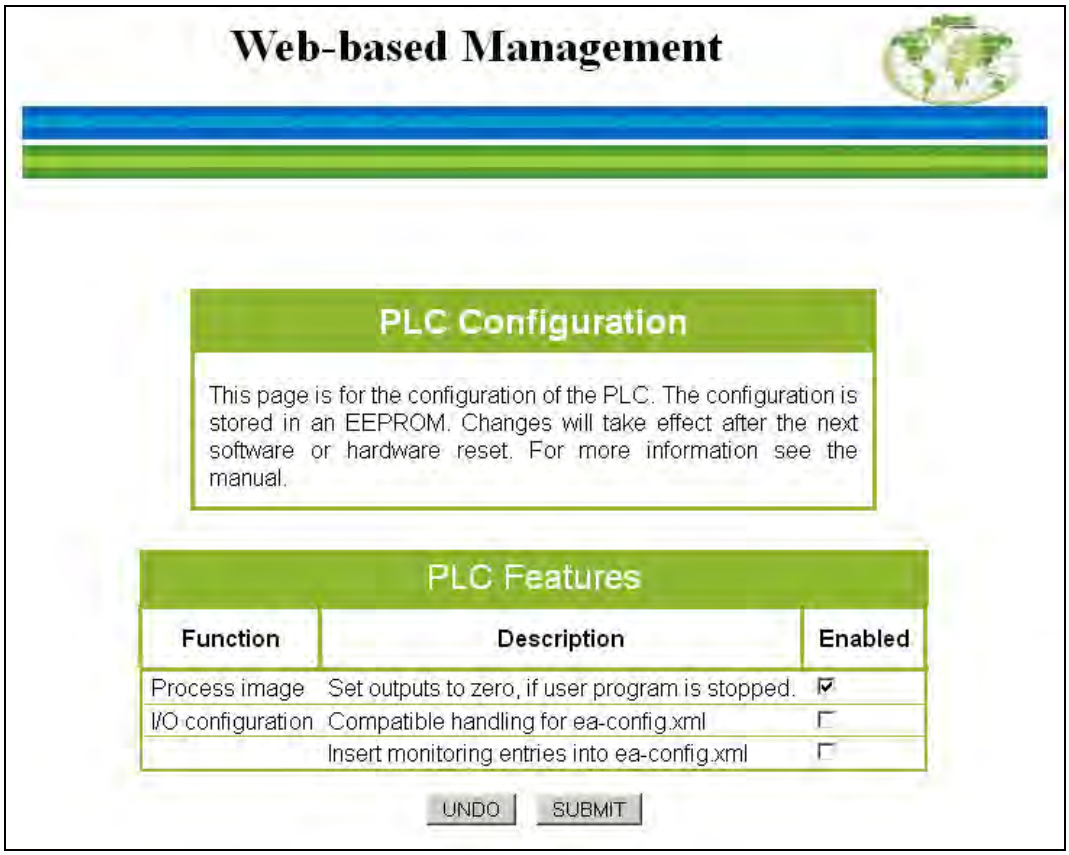

Web-based Management
PLC Configuration
This page is for the configuration of the PLC. The configuration is stored in an EEPROM. Changes will take effect after the next software or hardware reset. For more information see the manual.
PLC Features
Function
Description
Enabled
Process image
Set outputs to zero, if user program is stopped.
I/O configuration
Compatible handling for ea-config.xml
Insert monitoring entries into ea-config.xml
UNDO
SUBMIT

Abbildung 64: WBM-Seite „PLC“

Tabelle 46: WBM-Seite „PLC“

<table>
<tr><th colspan="4">PLC Features</th></tr>
<tr><th>Funktion</th><th>Standardwert</th><th colspan="2">Beschreibung</th></tr>
<tr><td rowspan="2">Process image</td><td rowspan="2">Set outputs to zero, if user program is stopped ☐</td><td colspan="2">☑ Aktivieren, wenn alle Ausgänge bei Stoppen des Anwenderprogramms auf null gesetzt werden sollen</td></tr>
<tr><td colspan="2">☐ Deaktivieren, wenn alle Werte bei Stoppen des Anwenderprogramms auf dem letzten aktuellen Wert verbleiben sollen</td></tr>
<tr><td rowspan="6">I/O configu-ration</td><td rowspan="5">Compatible handling for ea-config.xml ☐</td><td colspan="2">☑ Aktivieren, wenn die Schreibberechtigungen auf die Ausgänge aller Busklemmen anhand einer vorhandenen Datei „ea-config.xml“ zugewiesen werden sollen.
Beachten Sie dabei, ob bereits eine Steuerungskonfiguration angelegt wurde und, wenn ja, ob diese korrekt oder fehlerhaft ist (siehe nachfolgende Tabelle).</td></tr>
<tr><td colspan="2">☐ Deaktivieren, wenn die Schreibberechtigungen auf die Ausgänge aller Busklemmen der SPS zugewiesen werden sollen. Beachten Sie dabei, ob bereits eine Steuerungskonfiguration angelegt wurde und, wenn ja, ob diese korrekt oder fehlerhaft ist (siehe nachfolgende Tabelle).</td></tr>
<tr><td></td><td>EA-Konfiguration (Funktion aktiviert)</td><td>EA-Konfiguration (Funktion deaktiviert, default)</td></tr>
<tr><td>Im Projekt ist <u>keine</u> Steuerungskon-figuration angelegt</td><td>Die Schreibberechtigungen auf die Ausgänge aller Module werden anhand einer vorhandenen ea-config.xml zugewiesen.
Die ea-config.xml muss in jeder Hinsicht fehlerfrei sein, sonst wird dem Standardfeldbus die Schreibberechtigung für alle Module zugewiesen.</td><td>Die Ausgänge aller Module werden der SPS zugewiesen.
Eine evtl. vorhandene ea-config.xml wird nicht berücksichtigt und überschrieben.</td></tr>
<tr><td>Im Projekt ist eine <u>korrekte</u> Steuerungskon-figuration angelegt</td><td colspan="2">Die Schreibberechtigung auf die Ausgänge der Module wird aus der Steuerungskonfiguration entnommen. Es wird eine entsprechende ea-config.xml im Filesystem erzeugt.</td></tr>
<tr><td>Im Projekt ist eine <u>falsche</u> Steuerungskon-figuration angelegt</td><td colspan="2">Der Standardfeldbus erhält die Schreibberechtigung auf die Ausgänge aller Module.</td></tr>
<tr><td rowspan="2">Insert monitoring entries into ea-config.xml ☐</td><td colspan="2">☑ Aktivieren, um auf der HTML-Seite „IO config“ für die angezeigten Datenkanäle zusätzlich auch die aktuellen Prozesswerte anzuzeigen.</td></tr>
<tr><td colspan="2">☐ Deaktivieren, wenn auf der HTML-Seite „IO config“ keine Prozesswerte angezeigt werden sollen.</td></tr>
</table>

10.12 Features

Auf der HTML-Seite „Features“ aktivieren bzw. deaktivieren Sie zusätzliche Funktionen.

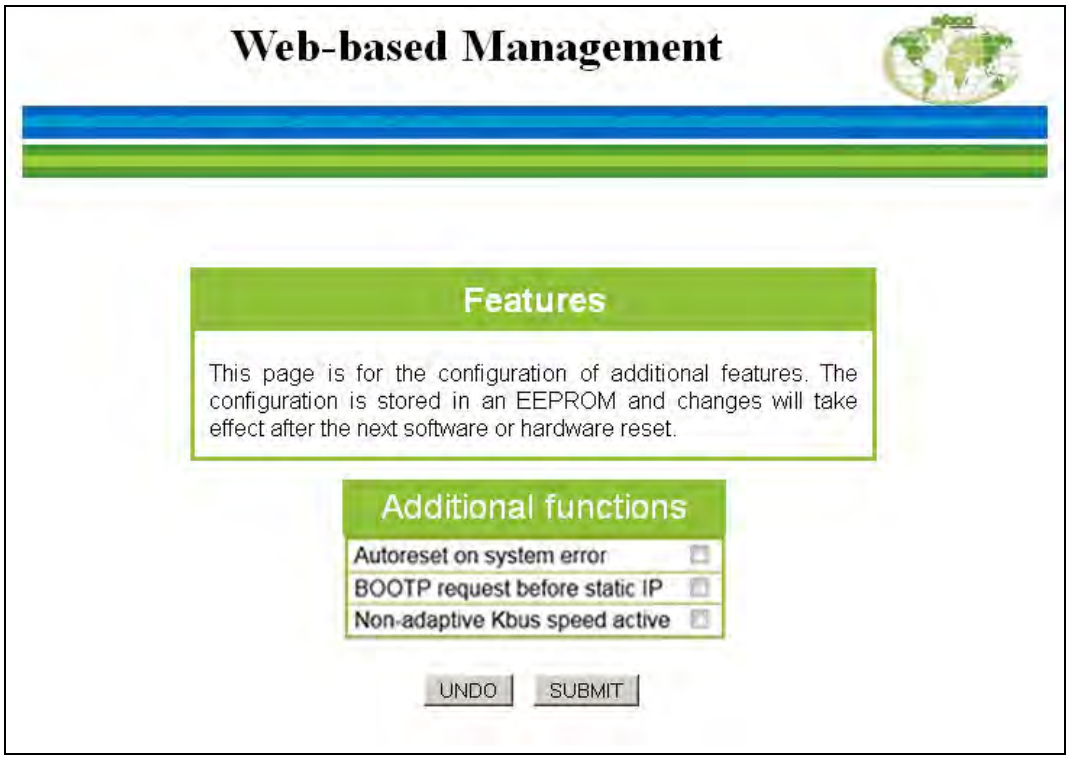

Web-based Management
Features
This page is for the configuration of additional features. The configuration is stored in an EEPROM and changes will take effect after the next software or hardware reset.
Additional functions
Autoreset on system error
BOOTP request before static IP
Non-adaptive Kbus speed active
UNDO
SUBMIT

Abbildung 65: WBM-Seite „Features“

Tabelle 47: WBM-Seite „Features“

Aditional functions		
Eintrag	**Standardwert**	**Beschreibung**
Autoreset on system error	☐	☑ Automatischer Software-Neustart beim Auftreten eines Systemfehlers aktiv
		☐ Automatischer Software-Neustart beim Auftreten eines Systemfehlers nicht aktiv
BootP Request before Static-IP	☐	☑ Automatisches Setzen der statischen IP-Adressen aktiv. Bei dieser Konfiguration verwendet der Feldbuskoppler/ -controller eine statisch konfigurierte IP-Adresse, falls die Anfrage über BootP fehl schlägt
		☐ Automatisches Setzen der statischen IP-Adressen nicht aktiv. Bei dieser Konfiguration wird die Anfrage der IP-Adresse über BootP im Falle eines Fehlers wiederholt.
Non-adaptive Kbus speed active	☐	☑ Deaktiviert die Geschwindigkeitsanpassung des Klemmenbusses und setzt die Pausenzeit statisch auf 14 ms. Dadurch verringert sich die Klemmenbusgeschwindigkeit, für die PLC-Applikation steht entsprechend mehr Rechenleistung zur Verfügung.
		☐ Aktiviert die Geschwindigkeitsanpassung des Klemmenbusses.

10.13 I/O Config

Auf der HTML-Seite „I/O Config“ sehen Sie eine Übersicht der Konfiguration bzw. der Schreibzugriffsrechte für die Ausgänge Ihres Feldbusknotens.

In dem Fenster wird der Knotenaufbau dargestellt, den Sie mit dem I/O-Konfigurator der WAGO-I/O-*PRO* erstellt haben. Werden keine Busklemmen angezeigt, haben Sie noch keine Hardware-Konfiguration und keine Zuweisung von Schreibzugriffsrechten vorgenommen. In diesem Fall werden entsprechend der Funktion „I/O configuration – Compatible handling for ea-config.xml“ (HTML-Seite „PLC“) die Schreibberechtigungen aller Ausgänge entweder dem Standardfeldbus oder der SPS zugewiesen.

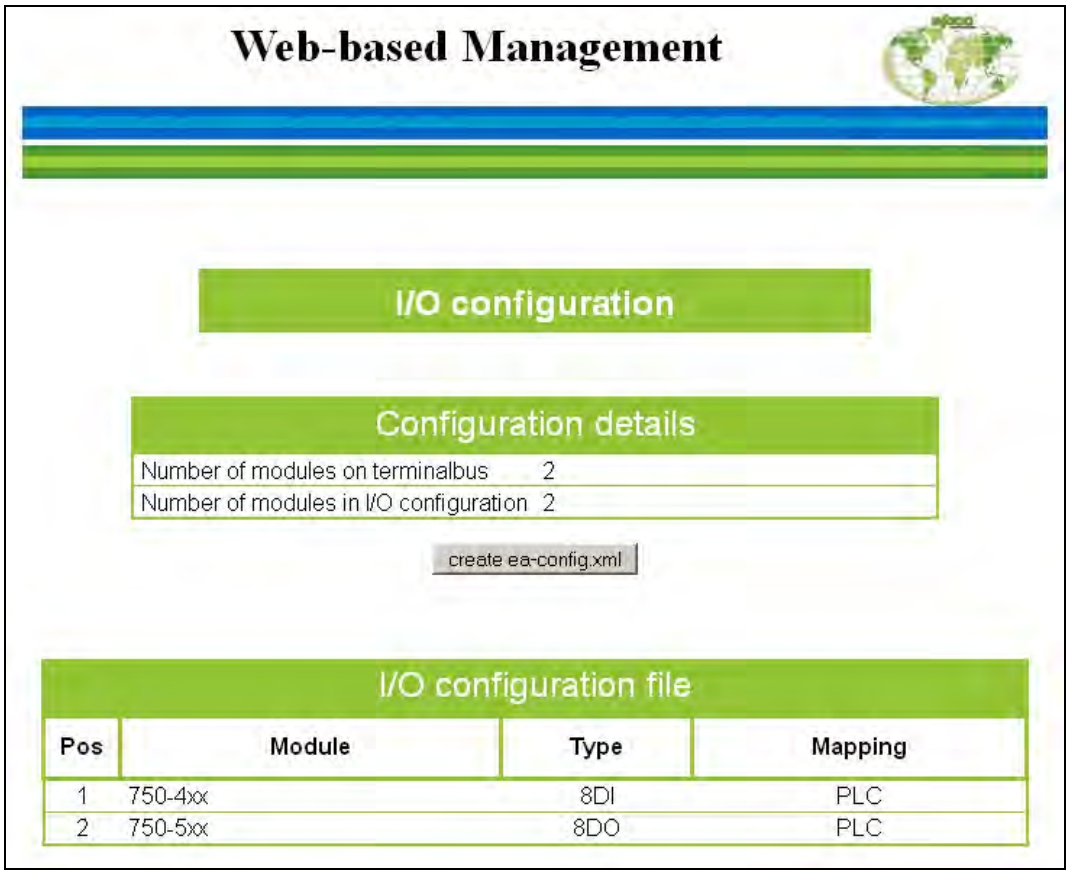


Abbildung 66: WBM-Seite „I/O Configuration“ (Beispielkonfiguration)

Information

Weitere Information

Detaillierte Informationen zu dem I/O-Konfigurator der WAGO-I/O-*PRO* finden Sie im Kapitel „In Betrieb nehmen“.

Ist auf der Webseite „PLC“ zusätzlich noch die Funktion „I/O configuration – Insert monitoring entries into ea-config.xml“ mit einem Haken ausgewählt/aktiviert, werden für die angezeigten Datenkanäle auch die aktuellen Prozesswerte angezeigt.

Tabelle 48: WBM-Seite „I/O Config“

Configuration details		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Number of modules on terminalbus	5	Anzahl der Busklemmen (Hardware)
Number of modules in I/O configuration	5	Anzahl der Busklemmen in der Hardware-Konfiguration des I/O-Konfigurators
[create ea-config.xml]	-	Schreibt den aktuellen Busklemmenaufbau und die Feldbuszuweisung in die Datei „ea-config.xml“.

I/O configuration file		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Pos	1	Position der Busklemme in der Hardware
Module	750-4xx M001Ch1 M001Ch2	Bestellnummer der eingebundenen Busklemme M = Module, 001 = Position 1, Ch1 = Kanal 1 M = Module, 002 = Position 2, Ch2 = Kanal 2
Type	2DI	Busklemmentyp, z. B. 2DI (2-Kanal-Digitaleingangsklemme)
Mapping	Fieldbus 3	Mapping über PLC, Fieldbus 1 etc. (Einträge sind koppler-/controllerabhängig, siehe in WAGO-I/O-*PRO* unter Steuerungsparameter/Modulparameter)

Hinweis

Busklemmen in den I/O-Konfigurator eintragen!
Tragen Sie Ihre verwendeten Busklemmen im I/O-Konfigurator von WAGO-I/O-*PRO* ein. Öffnen Sie dazu im Register **Ressourcen** die **Steuerungskonfiguration** und fügen Sie Ihre Busklemmen der Klemmenbusabbildung hinzu.
Die hinzugefügten Busklemmen müssen in Reihenfolge und Anzahl mit Ihrer Hardware übereinstimmen. Als Kontrolle dienen die Einträge „Number of modules on terminalbus“ und „Number of modules in I/O configuration“ auf der HTML-Seite „I/O Config“.

10.14 Disk Info

Auf der Seite „Disk Info“ werden Informationen zum Flash-Dateisystem angezeigt.

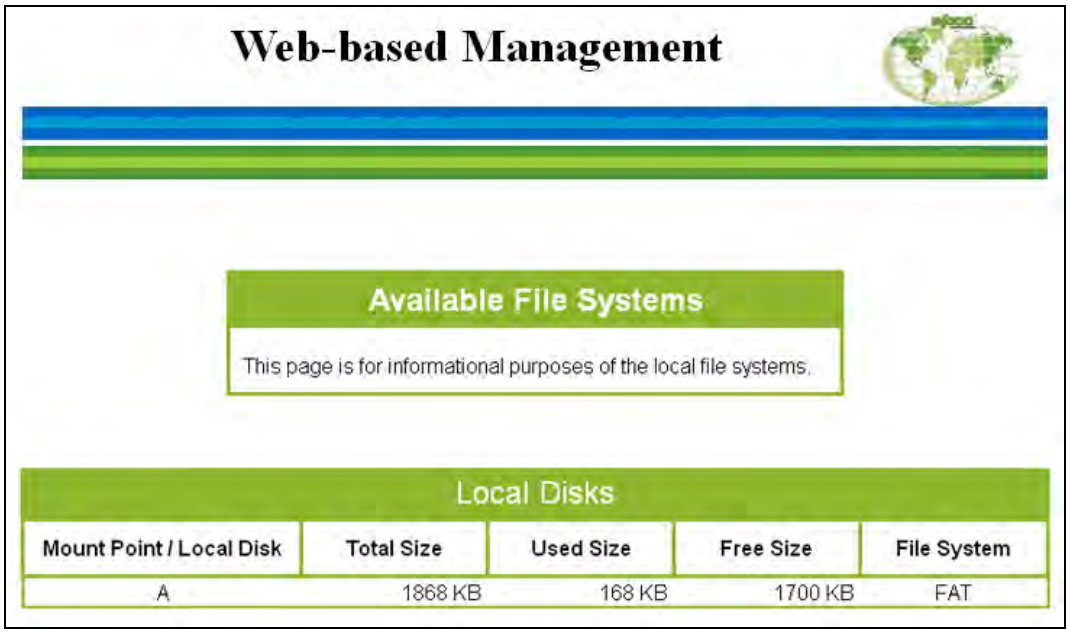


Abbildung 67: WBM-Seite „Disk Info“

Tabelle 49: WBM-Seite „Disk Info“

Local Disks		
Eintrag	**Wert (Beispiel)**	**Beschreibung**
Mount Point/Local Disk	A	Verzeichnis
Total Size	1868 KB	Gesamtgröße des Dateisystems
Used Size	172 KB	Belegter Speicherplatz
Free Size	1696 KB	Freier Speicherplatz
File System	FAT	Dateisystem (File Allocation Table)

11 Diagnose

11.1 LED-Signalisierung

Für die Vor-Ort-Diagnose besitzt der Feldbuscontroller LEDs, die den Betriebszustand des Feldbuskopplers/-controllers bzw. des ganzen Knotens anzeigen (siehe folgende Abbildung).

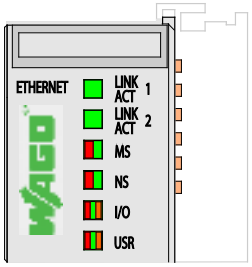


Abbildung 68: Anzeigeelemente

Die Diagnoseanzeigen und deren Bedeutung werden in den nachfolgenden Kapiteln erläutert.

Die LEDs sind gruppenweise den verschiedenen Diagnosebereichen zugeordnet:

Tabelle 50: LED-Zuordnung für die Diagnose

Diagnosebereich	LEDs
Feldbusstatus	• LINK ACT Port 1 • LINK ACT Port 2 • MS • NS
Knotenstatus	• I/O • USR

11.1.1 Feldbusstatus auswerten

Der Betriebszustand der Kommunikation über den Feldbus wird über die obere LED-Gruppe signalisiert, 'LINK ACT 1, 2', 'MS', und 'NS'.

Die zweifarbigen LEDs 'MS' (Module Status) und 'NS' (Network Status) werden ausschließlich vom ETHERNET/IP-Protokoll verwendet. Die Anzeigen dieser beiden LEDs entsprechen den ETHERNET/IP-Spezifikationen.

Tabelle 51: Diagnose des Feldbusstatus – Abhilfe im Fehlerfall

LED-Status	Bedeutung	Abhilfe
LINK ACT 1, 2		
grün	Der Feldbusknoten hat Verbindung zu dem physikalischen Netzwerk.	-
grün blinkend	Der Feldbusknoten sendet oder empfängt Ethernet-Telegramme	-
aus	Der Feldbusknoten hat keine Verbindung zu physikalischem Netzwerk.	1. Überprüfen Sie das Feldbuskabel.
MS		
grün	Das System arbeitet einwandfrei.	-
grün blinkend	Das System ist noch nicht konfiguriert.	-
rot	Das System zeigt einen nicht behebbaren Fehler an.	1. Führen Sie einen Neustart des Gerätes durch, indem Sie die Versorgungs-spannung aus- und einschalten. 2. Sollte der Fehler weiterhin bestehen, wenden Sie sich an den WAGO-I/O-Support.
rot/grün blinkend	Selbsttest	-
aus	Es ist keine Betriebsspannung für das System vorhanden.	1. Überprüfen Sie die Stromversorgung.
NS		
grün	Mindestens eine Verbindung (MODBUS/ TCP oder ETHERNET/ IP) ist aufgebaut (auch Verbindung zum Message-Router gilt)	-
grün blinkend	Es besteht keine Verbindung (MODBUS/TCP oder ETHERNET/IP).	-
rot	Das System hat eine doppelt verwendete IP-Adresse erkannt.	1. Verwenden Sie eine noch nicht verwendete IP-Adresse.
rot blinkend	Mindestens eine Verbindung (MODBUS/ TCP oder ETHERNET/ IP) hat einen Timeout gemeldet, bei welchem das Gerät als Target fungiert.	1. Starten Sie das Gerät durch Aus- und Einschalten der Versorgungsspannung neu. 2. Bauen Sie die Verbindung erneut auf.
rot/grün blinkend	Selbsttest	-
aus	Dem System ist keine IP-Adresse zugeordnet.	1. Ordnen Sie dem System z. B. über BootP oder DHCP eine IP-Adresse zu.

11.1.2 Knotenstatus auswerten – I/O-LED (Blinkcode-Tabelle)

Der Betriebszustand der Kommunikation zwischen dem Feldbuskoppler/ -controller und den Busklemmen wird über die I/O-LED signalisiert.

Tabelle 52: Diagnose des Knotenstatus – Abhilfe im Fehlerfall

LED-Status	Bedeutung	Abhilfe
I/O		
grün	Datenzyklus auf dem Klemmenbus.	Normale Betriebsbedingung
orange blinkend	Start der Firmware. Der Anlauf wird durch ca. 1 … 2 Sekunden schnelles Blinken angezeigt.	-
rot dauerhaft	Es liegt ein Hardware-Defekt des Feldbuskopplers/-controllers vor.	Tauschen Sie den Feldbuskoppler/ -controller aus.
rot blinkend	Blinken mit ca. 10 Hz weist auf die Initialisierung des Klemmenbusses oder auf einen allgemeinen Klemmenbusfehler hin.	Beachten Sie nachfolgenden Blinkcode.
rot zyklisch blinkend	Es werden auftretende Klemmenbusfehler mit bis zu drei nacheinander folgende Blinksequenzen angezeigt. Zwischen diesen Sequenzen ist jeweils eine kurze Pause.	Werten Sie die angezeigten Blink-sequenzen anhand der nachfolgenden Blinkcode-Tabelle aus. Das Blinken zeigt eine Fehlermeldung an, die sich aus einem Fehlercode und einem Fehlerargument zusammensetzt.
aus	Kein Datenzyklus auf dem Klemmenbus.	Die Versorgungsspannung des Feldbuskopplers/-controllers ist nicht eingeschaltet.

Nach Einschalten der Versorgungsspannung läuft das Gerät hoch. Dabei blinkt die I/O-LED orange.

Anschließend wird der Klemmenbus initialisiert. Dies wird durch rotes Blinken mit 10 Hz für 1 … 2 Sekunden signalisiert.

Nach fehlerfreier Initialisierung zeigt die I/O-LED grünes Dauerlicht.
Im Fehlerfall blinkt die I/O-LED rot.

Mit Hilfe eines Blinkcodes werden detaillierte Fehlermeldungen angezeigt. Ein Fehler wird über bis zu 3 Blinksequenzen zyklisch dargestellt.

Nach Beseitigung eines Fehlers ist der Feldbusknoten durch Aus- und Einschalten der Versorgungsspannung des Gerätes neu zu starten.

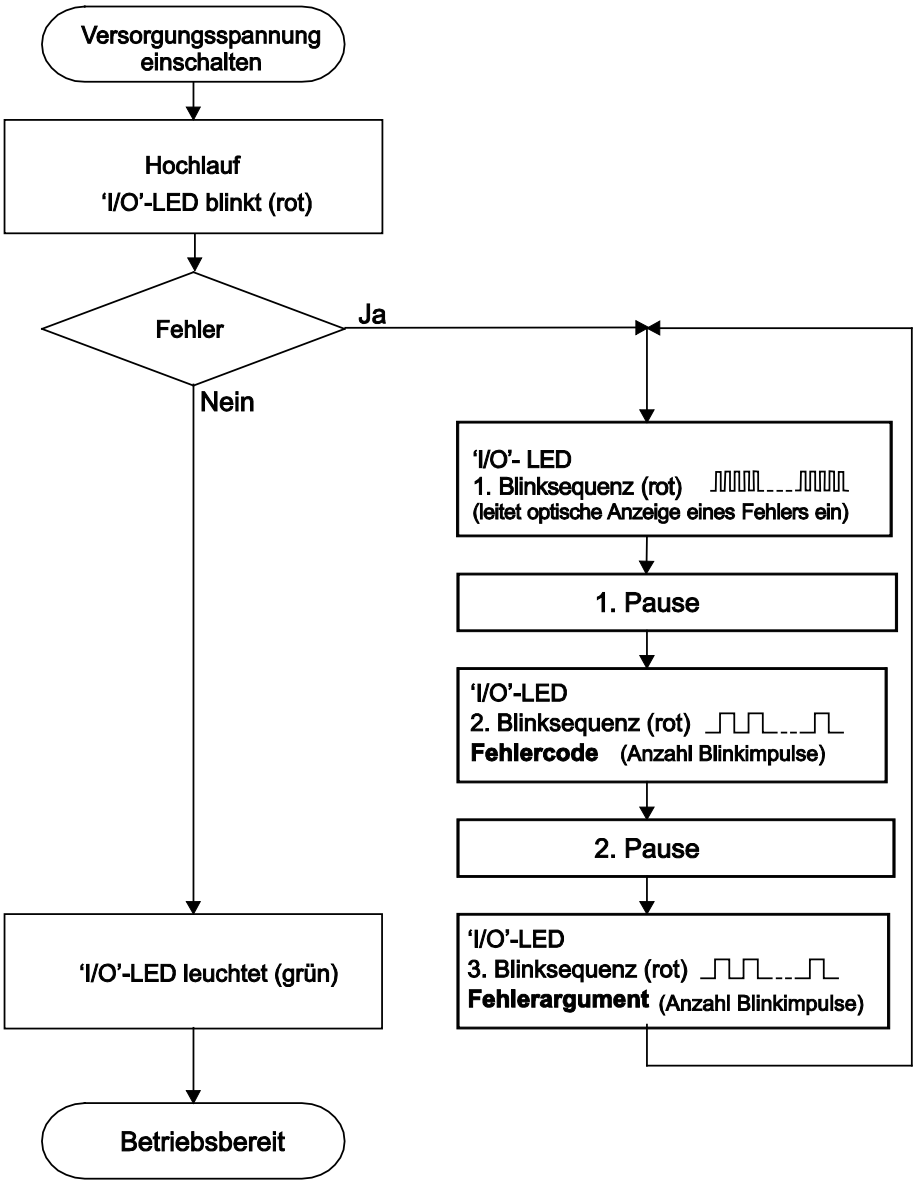


Abbildung 69: Knotenstatus -Signalisierung der I/O-LED

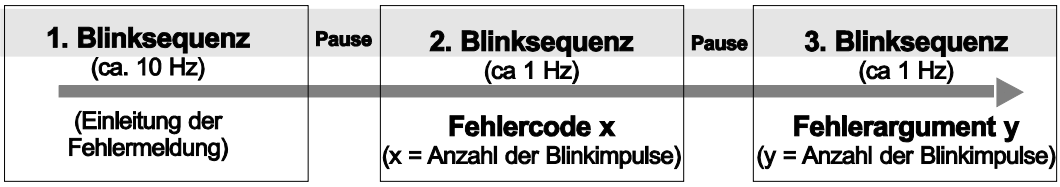


Abbildung 70: Kodierung der Fehlermeldung

Beispiel eines Klemmenfehlers:

- Die I/O-LED leitet mit der 1. Blinksequenz (ca. 10 Hz) die Fehleranzeige ein.

- Nach der ersten Pause folgt die 2. Blinksequenz (ca. 1 Hz):
  Die I/O-LED blinkt viermal.
  Damit wird der Fehlercode 4 „Datenfehler Klemmenbus“ signalisiert.

- Nach der zweiten Pause folgt die 3. Blinksequenz (ca. 1 Hz):
  Die I/O-LED blinkt zwölf Mal.
  Das Fehlerargument 12 bedeutet, dass der Klemmenbus nach der 12. Busklemme unterbrochen ist.

Somit ist die 13. Busklemme entweder defekt oder aus dem Verbund herausgezogen.

Tabelle 53: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 1

Fehlercode 1: "Hardware- und Konfigurationsfehler"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
1	Interner Speicherüberlauf bei Inlinecode-Generierung.	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. Sollte der Fehler weiterhin bestehen, tauschen Sie den Feldbuscontroller aus.
2	Busklemme(n) mit nicht unterstützter Datenstruktur	1. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 2. Stecken sie die Endklemme in die Mitte des Knotens. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. --- Blinkt die LED weiter? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). --- Blinkt die LED nicht? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 5. Schalten Sie die Versorgungsspannung wieder ein. 6. Wiederholen Sie den im Schritt 4 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 7. Tauschen Sie die fehlerhafte Busklemme aus. 8. Erkundigen Sie sich nach einem Firmware-Update für den Feldbuscontroller.
3	Ungültige Prüfsumme im Parameterbereich des Feldbuscontrollers	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
4	Fehler beim Schreiben in das serielle EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
5	Fehler beim Lesen aus dem seriellen EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.

Tabelle 53: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 1

Fehlercode 1: "Hardware- und Konfigurationsfehler"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
6	Die ermittelte Busklemmen-Konfiguration nach einem Klemmenbus-Reset (AUTORESET) differiert zu der, die beim letzten Hochlauf des Feldbuscontrollers ermittelt wurde.	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu.
7	Ungültige Hardware-Firmware-Kombination	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
8	Zeitüberschreitung beim Zugriff auf das serielle EEPROM	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
9	Buscontroller Initialisierungsfehler	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung wieder ein.
10 … 13	nicht genutzt	
14	Maximale Anzahl an Gateway- bzw. Mailbox-Busklemmen überschritten	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der entsprechenden Busklemmen auf ein zulässiges Maß. 3. Schalten Sie die Versorgungsspannung wieder ein.

Tabelle 54: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 2

Fehlercode 2: „Prozessabbildüberschreitung“		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
1	nicht genutzt	-
2	Maximale Prozessabbildgröße überschritten	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen. 3. Schalten Sie die Versorgungsspannung ein.

Tabelle 55: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 3

Fehlercode 3: „Protokollfehler Klemmenbus“		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	Klemmenbus-kommunikation gestört, fehlerhafte Baugruppe ist nicht identifizierbar	--- Befinden sich Potentialeinspeiseklemmen mit Busnetzteil (750-613) im Knoten? --- 1. Überprüfen Sie, ob diese Busklemmen korrekt mit Spannung versorgt werden. 2. Entnehmen Sie dieses dem Zustand der zugehörigen Status-LEDs. --- Sind alle Busklemmen ordnungsgemäß angeschlossen oder befinden sich keine Busklemmen vom Typ 750-613 im Knoten? --- 1. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 2. Stecken Sie die Endklemme in die Mitte des Knotens. 3. Schalten Sie die Versorgungsspannung wieder ein. 4. --- Blinkt die LED weiter? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). --- Blinkt die LED nicht? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 5. Schalten Sie die Versorgungsspannung wieder ein. 6. Wiederholen Sie den im Schritt 4 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 7. Tauschen Sie die fehlerhafte Busklemme aus. 8. Befindet sich nur noch eine Busklemme am Feldbus-controller und die LED blinkt, ist entweder diese Busklemme defekt oder der Feldbuscontroller. 9. Tauschen Sie die defekte Komponente.

Tabelle 56: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 4

Fehlercode 4: „Physikalischer Fehler Klemmenbus“		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	Fehler bei der Klemmenbus-datenübertragung oder Unterbrechung des Klemmenbusses an dem Feldbuscontroller	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Stecken Sie eine Busklemme mit Prozessdaten hinter den Feldbuscontroller. 3. Schalten Sie die Versorgungsspannung ein. 4. Beobachten Sie das signalisierte Fehlerargument. ---Wird kein Fehlerargument auf der I/O-LED ausgegeben?--- 5. Tauschen Sie den Feldbuscontroller aus. --- Wird ein Fehlerargument auf der I/O-LED ausgegeben? --- 5. Ermitteln Sie die fehlerhafte Busklemme, indem Sie die Versorgungsspannung ausschalten. 6. Stecken Sie die Endklemme in die Mitte des Knotens. 7. Schalten Sie die Versorgungsspannung wieder ein. 8. - Blinkt die LED weiter? - Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der ersten Hälfte des Knotens (zum Feldbuscontroller hin). --- Blinkt die LED nicht? --- Schalten Sie die Versorgungsspannung aus, und stecken Sie die Endklemme in die Mitte der zweiten Hälfte des Knotens (vom Feldbuscontroller weg). 9. Schalten Sie die Versorgungsspannung wieder ein. 10. Wiederholen Sie den im Schritt 6 beschriebenen Vorgang mit halbierten Schrittweiten, bis die fehlerhafte Busklemme gefunden ist. 11. Tauschen Sie die fehlerhafte Busklemme aus. 12. Befindet sich nur noch eine Busklemme am Feldbus-controller und die LED blinkt, ist entweder diese Busklemme defekt oder der Feldbuscontroller. 13. Tauschen Sie die defekte Komponente.
n*	Es liegt eine Klemmenbus-unterbrechung hinter der n-ten Busklemme mit Prozessdaten vor, die maximal unterstützte Anzahl ist erreicht, die nachfolgenden werden nicht mehr unterstützt.	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen bis zur n-ten Busklemme mit Prozessdaten. 3. Schalten Sie die Versorgungsspannung ein.

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

Tabelle 57: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 5

Fehlercode 5: „Initialisierungsfehler Klemmenbus“		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
n*	Fehler bei der Registerkom-munikation während der Klemmenbus-Initialisierung	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie die (n+1)-te Busklemme mit Prozessdaten aus. 3. Schalten Sie die Versorgungsspannung ein.

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

Tabelle 58: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 6

Fehlercode 6: " Projektierungsfehler Knotenkonfiguration"		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
1	Ungültige MAC-ID	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie den Feldbuscontroller aus. 3. Schalten Sie die Versorgungsspannung ein.
2	Initialisierungsfehler ETHERNET-Hardware	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
3	Initialisierungsfehler TCP/IP-Stack	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
4	Konfigurationsfehler Netzwerk (keine IP-Adresse)	1. Überprüfen Sie die Einstellungen des BootP-Servers.
5	Fehler bei der Initialisierung eines Applikationsproto-kolls	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
6	Maximale Prozessabbildgröße überschritten	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen.
7	IP-Adresse des Feldbuscontrollers ist mehrfach im Netzwerk vorhanden	1. Ändern Sie die Konfiguration: Verwenden Sie eine noch nicht im Netz vorhandene IP-Adresse. 2. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu.
8	Fehler beim Erstellen des Prozessabbildes	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Reduzieren Sie die Anzahl der Busklemmen. 3. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 4. Wird der Fehler weiterhin gemeldet? Tauschen Sie den Feldbuscontroller aus.
9	Fehler beim Mappen der Busklemmen zu einem Feldbus	1. Überprüfen Sie die Datei EA-Config.xml auf Ihrem Feldbuscontroller.

Tabelle 59: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 7

Fehlercode 7: „Nicht unterstützte Busklemme“		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
n	Erste nicht unterstützte Busklemme an Stelle n	1. Schalten Sie die Versorgungsspannung des Knotens aus. 2. Tauschen Sie die n-te Busklemme mit Prozessdaten aus bzw. reduzieren Sie die Anzahl der Busklemmen auf n-1. 3. Schalten Sie die Versorgungsspannung ein.

Tabelle 60: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 8...9

Fehlercode 8... 9 – nicht genutzt –		
Fehler-argument	**Fehler-beschreibung**	**Abhilfe**
-	nicht genutzt	-

Tabelle 61: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 10

Fehlercode 10: "Fehler bei der SPS-Programmbearbeitung"		
Fehler-argument	**Fehlerbeschreibung**	**Abhilfe**
1	Fehler beim Aufsetzen des PFC-Laufzeitsystems	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Sollte der Fehler weiterhin gemeldet werden, wenden Sie sich an den I/O-Support.
2	Fehler beim Generieren des PFC-Inline-Codes	1. Starten Sie den Feldbuscontroller durch Aus- und Einschalten der Versorgungsspannung neu. 2. Sollte der Fehler weiterhin gemeldet werden, wenden Sie sich an den I/O-Support.
3	Eine IEC-Task hat die maximale Laufzeit überschritten oder das Aufrufintervall der IEC-Task konnte nicht eingehalten werden (Zeitüberwachung)	1. Überprüfen Sie die Task-Konfiguration bezüglich der eingestellten Aufrufintervalle und Überwachungszeiten.
4	nicht genutzt	
5	Fehler beim Abgleich der Steuerungs-konfiguration mit dem Klemmenbus	1. Überprüfen Sie die Angabe der gesteckten Klemmen in der CODESYS-Steuerungskonfiguration 2. Gleichen Sie diese mit den tatsächlich gesteckten Klemmen ab. 3. Übersetzen Sie das Projekt erneut. 4. Bringen Sie das Projekt wieder auf den Feldbuscontroller.

Tabelle 62: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 11

Fehlercode 11: "Gateway-/Mailbox-Klemmen Fehler"		
Fehler-argument	**Fehlerbeschreibung**	**Abhilfe**
1	Es sind zu viele Gateway-Klemmen gesteckt	1. Vermindern Sie die Zahl der Gateway-Klemmen
2	Maximale Mailbox-Größe überschritten	1. Verkleinern Sie die Mailbox-Größe
3	Maximale PA-Größe überschritten aufgrund von gesteckten Gateway-Klemmen	1. Verkleinern Sie die Datenbreite der Gateway-Klemmen

* Die Anzahl der Blinkimpulse (n) zeigt die Position der Busklemme an.
Busklemmen ohne Daten werden nicht mitgezählt (z. B. Einspeiseklemme ohne Diagnose)

11.1.2.1 USR-LED

Für die visuelle Ausgabe von Informationen steht dem Anwender die unterste Anzeige LED („USR“) zur Verfügung.

Die Ansteuerung der LED aus dem Anwenderprogramm erfolgt mit den Funktionen aus der WAGO-I/O-*PRO*-Bibliothek „Visual.lib“.

11.2 Fehlerverhalten

11.2.1 Feldbusausfall

Ein Feldbus- und damit ein Verbindungsausfall liegt vor, wenn die eingestellte Reaktionszeit des Watchdogs ohne Anstoß durch die übergeordnete Steuerung abgelaufen ist. Dies kann beispielsweise passieren, wenn der Master abgeschaltet oder das Buskabel unterbrochen ist. Auch ein Fehler im Master kann zum Feldbusausfall führen. Es ist keine Verbindung über ETHERNET gegeben.

Der MODBUS-Watchdog überwacht die über das MODBUS-Protokoll laufende MODBUS-Kommunikation. Sofern der MODBUS-Watchdog konfiguriert und aktiviert wurde, wird ein Feldbusausfall durch das Leuchten der roten I/O-LED angezeigt.

Eine protokollunabhängige Feldbusüberwachung ist über den Funktionsblock 'FBUS_ERROR_INFORMATION' der Bibliothek 'Mod_com.lib' möglich, der die physikalische Verbindung zwischen Busklemmen und Feldbuscontroller überprüft und die Auswertung der Watchdog-Register im Steuerungsprogramm übernimmt. Der Klemmenbus bleibt funktionsfähig und die Prozessabbilder bleiben erhalten. Das Steuerungsprogramm kann autark abgearbeitet werden.

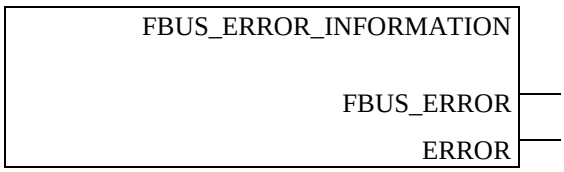


Abbildung 71: Funktionsblock zur Ermittlung des Feldbusausfalls

'FBUS_ERROR' (BOOL)	= FALSE	= kein Fehler
	= TRUE	= Feldbusausfall
'ERROR' (WORD)	= 0	= kein Fehler
	= 1	= Feldbusausfall

Mit Hilfe dieser Funktionsblockausgänge und einem entsprechend programmierten Steuerungsprogramm kann der Knoten bei Feldbusausfall in einen sicheren Zustand geführt werden.

Information

Feldbusausfallerkennung über das MODBUS-Protokoll:
Detaillierte Informationen zu dem Watchdog-Register entnehmen Sie dem Kapitel „MODBUS-Funktionen“, „Watchdog (Verhalten bei Feldbusausfall)“.
Protokollunabhängige Feldbusausfall-Erkennung:
Die Bibliothek 'Mod_com.lib' mit dem Funktionsblock 'FBUS_ERROR_INFORMATION' ist standardmäßig im Setup der WAGO-I/O-*PRO* enthalten. Sie binden die Bibliothek über das Register „Ressourcen“ links unten auf der Arbeitsfläche ein. Klicken Sie auf **Einfügen** und **weitere Bibliotheken**. Die Mod_com.lib befindet sich im Ordner C:\Programme\WAGO Software\CODESYS V2.3\Targets\WAGO\Libraries\32_Bit

11.2.2 Klemmenbusfehler

Ein Klemmenbusfehler wird über die I/O-LED angezeigt.

I/O-LED blinkt rot:
Bei einem Klemmenbusfehler erzeugt der Feldbuscontroller eine Fehlermeldung (Fehlercode und Fehlerargument).
Ein Klemmenbusfehler entsteht beispielsweise durch eine herausgezogene Busklemme.
Wenn dieser Fehler während des Betriebes auftritt, verhalten sich die Ausgangsklemmen wie beim Klemmenbusstopp.
Wenn der Klemmenbusfehler behoben ist, läuft der Feldbuscontroller nach einem Aus- und Einschalten wie beim Betriebsstart hoch. Die Übertragung der Prozessdaten wird wieder aufgenommen und die Ausgänge im Knoten werden entsprechend gesetzt.

Soll in dem Steuerungsprogramm der Funktionsbaustein 'KBUS_ERROR_INFORMATION' ausgewertet werden, dann sind die Ausgangswerte 'ERROR','BITLEN', 'TERMINALS' und 'FAILADDRESS' relevant.

'ERROR'	= FALSE	= kein Fehler
('BITLEN'		= Bitlänge des Klemmenbus-Schieberegisters
'TERMINALS'		= Anzahl der gesteckten Busklemmen)

'ERROR'	= TRUE	= Klemmenbusfehler
('BITLEN'		= 0
'TERMINALS'		= 0)
'FAILADRESS'		= Position der Busklemme, nach der die Klemmenbusunterbrechung aufgetreten ist, analog zu dem ausgeblinkten Fehlerargument der I/O-LED)

12 Feldbuskommunikation

Die Feldbuskommunikation zwischen Master-Anwendung und einem auf dem ETHERNET-Standard basierenden WAGO-Feldbuskoppler/-controller findet in der Regel über ein feldbusspezifisch implementiertes Anwendungsprotokoll statt.

Je nach Anwendung, kann dieses z. B. MODBUS/TCP (UDP), EtherNet/IP, BACnet/IP, KNX IP, PROFINET, sercos oder sonstiges sein.

Hinzu kommen zu dem ETHERNET-Standard und dem feldbusspezifischen Anwendungsprotokoll außerdem noch einige, für eine zuverlässige Kommunikation und Datenübertragung wichtige Kommunikationsprotokolle und darauf aufbauend noch weitere Protokolle für die Konfiguration und Diagnose des Systems, die in den ETHERNET basierenden WAGO-Feldbuskoppler/-controller implementiert sind.

Diese Protokolle werden in den weiteren Kapiteln näher erläutert.

12.1 Implementierte Protokolle

12.1.1 Kommunikationsprotokolle

12.1.1.1 IP (Internet Protocol)

Das Internet-Protokoll (IP) teilt Datentelegramme in Segmente und ist verantwortlich für deren Beförderung von einem Netzteilnehmer zu einem anderen. Die beteiligten Stationen können sich dabei in demselben Netzwerk befinden oder in verschiedenen physikalischen Netzwerken, die aber mit Routern miteinander verbunden sind.

Die Router sind in der Lage, verschiedene Pfade (Netzwerkübertragungswege) durch einen Netzwerkverbund auszuwählen und somit Überlastungen und Störungen einzelner Netze zu umgehen.

Dabei kann es jedoch vorkommen, dass einzelne Strecken gewählt werden, die kürzer sind als andere. Daraufhin können sich Telegramme überholen und die Reihenfolge (Sequenz) der Datenpakete ist falsch.

Die Gewährleistung der korrekten Übertragung muss deshalb in höheren Schichten, z. B. durch TCP erfolgen.

IP-Datenpaket

Die IP-Datenpakete enthalten neben den zu transportierenden Nutzdaten eine Fülle von Adress- und Zusatzinformationen in dem „Paketkopf“.

Tabelle 63: IP-Datenpaket

IP-Header	IP-Nutzdatenbereich

Die wichtigsten Informationen in dem IP-Header sind die IP-Adressen vom Absender und Empfänger sowie das benutzte Transportprotokoll.

IP-Adressen

Für die Kommunikation im Netz muss jeder Feldbusknoten über eine 32-Bit lange Internet-Adresse (IP Adresse) verfügen.

Hinweis

IP-Adressen müssen eindeutig sein!
Zum fehlerfreien Betrieb muss die eingestellte IP-Adresse im gesamten Netzwerk eindeutig sein, es darf nicht zwei Mal dieselbe IP-Adresse vergeben werden.

Wie unten aufgezeigt, gibt es verschiedene Adressklassen mit unterschiedlich langer Netzwerk-Identifikation (Net-ID) und Host-Rechner-Identifikation (Host-ID).
Die Net-ID definiert das Netzwerk, in dem sich der Teilnehmer befindet.
Die Host-ID identifiziert einen bestimmten Teilnehmer innerhalb dieses Netzwerkes.

Zur Adressierung werden Netze in mehrere Netzwerkklassen unterteilt:

- **Class A:** (Net-ID: Byte 1, Host-ID: Byte 2… Byte 4)

Tabelle 64: Netzwerkklasse Class A

z. B.	101	. 16	. 232	. 22
01100101		00010000	11101000	00010110
0	Net-ID	Host-ID		

Das höchste Bit bei Class A-Netzen ist immer '0'.
D. h., das höchste Byte kann im Bereich von '0 0000000' bis '0 1111111' liegen.
Der Adressbereich der Class A-Netze liegt somit im ersten Byte immer zwischen 0 und 127.

- **Class B:** (Net-ID: Byte 1 … Byte 2, Host-ID: Byte 3… Byte 4)

Tabelle 65: Netzwerkklasse Class B

z. B.	181	. 16	. 232	. 22
10110101		00010000	11101000	00010110
10	Net-ID		Host-ID	

Die höchsten Bits bei Class B-Netzen sind immer '10'.
D. h., das höchste Byte kann im Bereich von '10 000000' bis '10 111111' liegen.
Der Adressbereich der Class B-Netze liegt somit im ersten Byte immer zwischen 128 und 191.

- **Class C:** (Net-ID: Byte 1 … Byte 3, Host-ID: Byte 4)

Tabelle 66: Netzwerkklasse Class C

z. B.	201	. 16	. 232	. 22
11000101		00010000	11101000	00010110
110	Net-ID			Host-ID

Die höchsten Bits bei Class C-Netzen sind immer '110'.
D. h., das höchste Byte kann im Bereich von '110 00000' bis '110 11111' liegen.
Der Adressbereich der Class C-Netze liegt somit im ersten Byte immer zwischen 192 und 223.

- **Weitere Netzwerkklassen (D, E):** werden für Sonderaufgaben verwendet.

Eckdaten

Tabelle 67: Eckdaten Class A, B und C

Netzwerkklasse	Adressbereich des Netzwerkteils	Mögliche Anzahl von	
		Netzen	Hosts pro Netz
Class A	0.XXX.XXX.XXX ... 127.XXX.XXX.XXX	128 (2^7)	Ca. 16 Millionen (2^{24})
Class B	128.000.XXX.XXX ... 191.255.XXX.XXX	Ca. 16 Tausend (2^{14})	Ca. 65 Tausend (2^{16})
Class C	192.000.000.XXX ... 223.255.255.XXX	Ca. 2 Millionen (2^{21})	254 (2^8)

Jedem ETHERNET basierenden Feldbuskoppler oder Feldbuscontroller kann über das implementierte BootP-Protokoll sehr leicht eine IP-Adresse zugeteilt werden. Als Empfehlung für ein kleines internes Netzwerk gilt, hier Netzwerkadressen aus dem Class C-Bereich zu wählen.

Hinweis

Bei IP-Adressen nicht 0.0.0.0 und 255.255.255.255 verwenden!
Beachten Sie, dass niemals alle Bits in einem Byte gleich ‚0' oder gleich ‚1' gesetzt sind (Byte = 0 oder 255). Diese sind für spezielle Funktionen reserviert und dürfen nicht vergeben werden. So darf z. B. darf die Adresse 10.0.10.10 wegen der 0 im zweiten Byte nicht verwendet werden.

Soll ein Netzwerk direkt mit dem Internet verbunden werden, so werden von einer zentralen Vergabestelle zugeteilte weltweit einmalige IP-Adressen verwendet. Die Vergabe in Deutschland erfolgt z. B. durch die DENIC eG (Deutsches Network Information Center) in Karlsruhe.

Hinweis

Internetanbindung nur durch autorisierten Netzwerkadministrator!
Beachten Sie, dass eine direkte Internetanbindung ausschließlich durch einen autorisierten Netzwerkadministrator erfolgen darf, deshalb ist eine solche Anbindung nicht in diesem Handbuch beschrieben.

Subnetzwerke

Um das Routing innerhalb von großen Netzwerken zu ermöglichen, wurde in der Spezifikation RFC 950 eine Konvention eingeführt. Dabei wird ein Teil der Internet-Adresse, die Host-ID, weiter unterteilt und zwar in eine Subnetzwerknummer und die eigentliche Stationsnummer des Knoten. Mit Hilfe der Netzwerknummer kann nun innerhalb des Teilnetzwerkes in interne Unternetzwerke verzweigt werden, von außen aber ist das gesamte Netzwerk als Einheit sichtbar. Größe und Lage der Subnetzwerk-ID sind nicht festgeschrieben, die Größe ist jedoch abhängig von der Anzahl der zu adressierenden Subnetze und die Anzahl der Hosts pro Subnetz.

Tabelle 68: Beispiel: Klasse B-Adresse mit Feld für Subnetzwerk-ID

1			8	16	24	32
1	0	...	Netz-ID	Subnetz-ID	Host-ID	

Subnetz-Maske

Für die Kodierung der Subnetze im Internet, wurde die sogenannte Subnetz-Maske eingeführt. Dabei handelt es sich um eine Bit-Maske, mit der spezielle Bits der IP-Adresse ausgeblendet bzw. selektiert werden können. Die Maske definiert, welche Bits der Host-ID für die Subnetz-Kodierung verwendet werden und welche die ID des Hosts bezeichnen.
Der gesamte IP-Adressbereich liegt theoretisch zwischen 0.0.0.0 und 255.255.255.255. Für die Subnetz-Maske sind jeweils die 0 und die 255 aus dem IP-Adressbereich reserviert.

Die von der jeweiligen Netzwerkklasse abhängigen Standard-Masken sehen wie folgt aus:

- **Class A-Subnetz-Maske:**

Tabelle 69: Subnetz-Maske für Class A-Netzwerke

255	.0	.0	.0

- **Class B-Subnetz-Maske:**

Tabelle 70: Subnetz-Maske für Class B-Netzwerke

255	.255	.0	.0

- **Class C-Subnetz-Maske:**

Tabelle 71: Subnetz-Maske für Class C-Netzwerke

255	.255	.255	.0

Je nach Subnetz-Unterteilung, können die Subnetz-Masken über 0 und 255 hinaus aber auch andere Werte enthalten, wie z. B. 255.255.255.128 oder 255.255.255.248, usw.

Die Subnetz-Masken-Nummer wird von dem Netzwerkadministrator zugewiesen. Zusammen mit der IP-Adresse bestimmt diese Nummer, zu welchem Netzwerk der PC und der Knoten gehört.

Der Empfängerknoten, der sich in einem Subnetz befindet, berechnet zunächst die richtige Netzwerknummer aus seiner eigenen IP-Adresse und der Subnetzwerk-Maske. Erst im Anschluss daran, überprüft er die Knotennummer und liest dann bei Übereinstimmung den gesamten Paket-Rahmen aus.

Tabelle 72: Beispiel für eine IP-Adresse aus einem Class B-Netz

IP-Adresse	172.16.233.200	10101100 00010000 11101001 11001000
Subnetz-Maske	255.255.255.128	11111111 11111111 11111111 10000000
Netz-ID	172.16.0.0	10101100 00010000 00000000 00000000
Subnetz-ID	0.0.233.128	00000000 00000000 11101001 10000000
Host-ID	0.0.0.72	00000000 00000000 00000000 01001000

Hinweis

Angabe der Netzwerk-Maske erforderlich!
Beachten Sie, dass die vom Administrator festgelegte Netzwerk-Maske bei der Installation des Netzwerkprotokolls genauso wie die IP-Adresse angegeben werden muss.

Gateway

Die Subnetze des Internets sind in der Regel über Gateways verbunden. Diese Gateways dienen dazu, Pakete an andere Netzwerke oder Subnetze weiterzuleiten. Für einen an das Internet angeschlossenen PC oder Feldbusknoten bedeutet das, dass zusätzlich zur IP-Adresse und Netzwerk-Maske für jede Netzwerkkarte die korrekte IP-Adresse des Standard-Gateways angegeben werden muss. Diese IP-Adresse sollte Ihnen ebenfalls von Ihrem Netzwerkadministrator zur Verfügung gestellt werden.
Ohne Angabe der IP-Adresse des Gateways bleibt die IP-Funktionalität auf das lokale Subnetz beschränkt.
Um direkt miteinander kommunizieren zu können, müssen Host und Gateway zum gleichen Subnetz gehören, d. h. dieselbe Netz-ID haben.

RAW-IP

Raw-IP kommt ohne Protokolle, wie z. B. PPP (Punkt-zu-Punkt-Protokoll) aus. Bei RAW-IP werden die TCP/IP-Pakete direkt, ohne Handshaking ausgetauscht, wodurch ein schnellerer Verbindungsaufbau möglich ist. Zuvor muss allerdings die Konfiguration mit einer festen IP-Adresse stattgefunden haben. Vorteile von RAW-IP sind eine hohe Datentransferrate und eine gute Stabilität.

IP-Multicast

Unter Multicast versteht man eine Übertragungsart von einem Punkt zu einer Gruppe, also eine Punkt-zu-Mehrpunkt-Übertragung oder auch Mehrpunktverbindung genannt. Der Vorteil von Multicast liegt darin, dass gleichzeitig Nachrichten über eine Adresse an mehrere Teilnehmer oder geschlossene Teilnehmergruppen (Closed User Groups) übertragen werden.
IP-Multicasting auf der Internetwork-Ebene wird durch das IGMP (Internet Group Message Protocol) realisiert; dieses Protokoll wird von Nachbar-Routern benutzt, um sich gegenseitig über Gruppenzugehörigkeiten zu informieren.
Bei der Verteilung von Multicast-Paketen im Subnetwork geht IP davon aus, dass der Datalink-Layer seinerseits Multicasting zur Verfügung stellt. Im Falle

ETHERNET sind Multicast-Adressen vorhanden, mit denen ein durch sie adressiertes Paket durch eine einzige Sendeoperation an mehrere Empfänger verschickt wird. Hier stützt man sich darauf, dass ein gemeinsames Medium die Möglichkeit bietet, Pakete an mehrere Empfänger gleichzeitig zu senden. Die Stationen untereinander müssen sich nicht informieren, wer zu einer Multicast-Adresse gehört – jede Station empfängt physikalisch jedes Paket.
Die Adressauflösung von IP-Adresse zu ETHERNET-Adresse wird algorithmisch gelöst, IP-Multicast-Adressen werden in ETHERNET-Multicastadressen eingebettet.

12.1.1.2 TCP (Transmission Control Protocol)

Aufgesetzt auf das Internet-Protokoll, übernimmt TCP (Transmission Control Protocol) die Sicherung des Datentransportes durch das Netzwerk.
Dazu stellt TCP für die Dauer der Datenübertragung eine Verbindung zwischen zwei Teilnehmern her. Die Kommunikation erfolgt im Voll-Duplexverfahren, d. h. beide Teilnehmer können gleichzeitig Daten empfangen und versenden.
Die übertragenen Nutzdaten werden von TCP mit einer 16bit-Prüfsumme versehen und jedes Datenpaket erhält eine Sequenznummer.
Der Empfänger überprüft anhand der Prüfsumme den korrekten Empfang des Paketes und verrechnet anschließend die Sequenznummer. Das Ergebnis nennt sich Acknowledgement-Nr. und wird mit dem nächsten selbst versendeten Paket als Quittung zurückgesendet. Dadurch ist gewährleistet, dass der Verlust von TCP-Paketen bemerkt wird, und diese im Bedarfsfall in korrekter Abfolge erneut gesendet werden können.

TCP-Datenpaket

Der Paketkopf eines TCP-Datenpaketes besteht aus mindestens 20 Byte und enthält unter anderem die Portnummer der Applikation des Absenders sowie die des Empfängers, die Sequenznummer und die Acknowledgement-Nr.
Das so entstandene TCP-Paket wird in den Nutzdatenbereich eines IP-Paketes eingesetzt, so dass ein TCP/IP-Paket entsteht.

TCP-Portnummern

TCP kann zusätzlich zur IP-Adresse (Netz- und Host-Adresse) gezielt eine spezielle Anwendung (Dienst) auf dem adressierten Host ansprechen. Dazu werden die auf einem Host befindlichen Anwendungen, wie z. B. Web-Server, FTP-Server und andere, über unterschiedliche Portnummern adressiert. Für bekannte Anwendungen werden feste Ports vergeben, auf die sich jede Anwendung beim Verbindungsaufbau beziehen kann.
(Beispiele: Telnet-Portnummer: 23, HTTP-Portnummer: 80).
Eine komplette Liste der „normierten Dienste“ findet sich in den Spezifikationen RFC 1700 (1994).

12.1.1.3 UDP (User Datagram Protocol)

Das UDP-Protokoll ist, wie auch das TCP-Protokoll, für den Datentransport zuständig. Im Vergleich zum TCP-Protokoll ist UDP nicht verbindungsorientiert. Das heißt es gibt keine Kontrollmechanismen bei dem Datenaustausch zwischen Sender und Empfänger. Der Vorteil dieses Protokolls liegt in der Effizienz der übertragenen Daten und damit in der resultierenden höheren Verarbeitungsgeschwindigkeit.

12.1.2 Konfigurations- und Diagnoseprotokolle

12.1.2.1 BootP (Bootstrap Protocol)

Mit dem „Bootstrap Protocol“ (BootP) können Sie dem Feldbuskoppler/ -controller in einem TCP/IP-Netzwerk eine IP-Adresse und andere Parameter zuweisen. Außerdem können Subnetzmaske und Gateway übermittelt werden. Die Protokollkommunikation besteht aus einer Client-Anfrage des Feldbuskopplers/-controllers und einer Server-Antwort von dem PC.

Über das Protokoll wird eine Broadcast-Anfrage auf Port 67 (BootP-Server) gesendet, welche die Hardware-Adresse (MAC-ID) des Feldbuskopplers/ -controllers enthält.

Der BootP-Server erhält die Nachricht. Er beinhaltet eine Datenbank, in dem MAC-ID und IP-Adressen einander zugeordnet sind. Wird die MAC-Adresse gefunden, wird eine Broadcast-Antwort über das Netz gesendet.

Der Feldbuskoppler/-controller lauscht auf dem vorgegebenen Port 68 auf die Antwort des BootP-Servers. Ankommende Pakete enthalten unter anderem die IP-Adresse und die MAC-Adresse des Feldbuskopplers/-controllers. An der MAC-Adresse erkennt ein Feldbuskoppler/-controller, ob die Nachricht für ihn bestimmt ist, und übernimmt bei Übereinstimmung die gesendete IP-Adresse in sein Netzwerk-Interface.

Hinweis

IP-Adressvergabe über BootP unter Windows und Linux möglich!
Sie können eine IP-Adresse mittels BootP-Server sowohl unter Windows- als auch unter Linux-Betriebssystemen vergeben.

Information

Weitere Information zur Adressvergabe mit BootP-Server
Die Vorgehensweise der Adressvergabe mit einem BootP-Server ist detailliert in dem Kapitel „Feldbusknoten in Betrieb nehmen“ beschrieben.

Der BootP-Client dient zum dynamischen Konfigurieren der Netzwerkparameter. Der ETHERNET TCP/IP-Feldbuscontroller besitzt einen BootP-Client, der neben der Standard-Option „IP-Adresse“ die folgenden Optionen unterstützt:

Tabelle 73: BootP-Optionen

Option	Bedeutung
[OPT1] Subnetzmaske	32 Bit Adressmaske, die anzeigt, welche Bits der IP-Adresse das Netzwerk und welche die Netzwerkstationen bestimmen.
[OPT2] Zeitzone	Zeitverschiebung zwischen der lokalen Zeit und der UTC (Universal Time Coordinated).
[OPT3] Gateway	IP-Adresse des Routers, der den Zugang zu anderen Netzwerken erlaubt.
[OPT6] DNS-Server	IP-Adresse der Name-Server, die einen Namen in eine IP-Adresse konvertieren. Es können bis zu 2 DNS-Server konfiguriert werden.
[OPT12] Hostname	Der Name des Hosts ist die eindeutige Bezeichnung eines Rechners in einem Netwerk. Der Hostname kann bis zu 32 Zeichen enthalten.
[OPT15] Domainename	Der Name der Domaine ist die eindeutige Bezeichnung eines Netzwerkes. Der Domainename kann bis zu 32 Zeichen enthalten.
[OPT42] NTP-Server	IP-Adresse des Network Time Servers. Ber der Vergabe eines NTP-Servers wird automatisch der SNTP-Client im Koppler aktiviert.

Über die WBM-Seite „Features“ kann ferner noch die Option “BootP Request before static IP” angewählt werden. Nach dem Neustart werden 5 BootP-Anfragen gesendet. Erfolgt auf keiner dieser Anfragen eine Antwort, versucht der Feldbuskoppler/-controller sich mit dem im EEPROM gespeicherten IP-Parametern zu konfigurieren.

Bei der Verwendung des Bootstrap-Protokolls zur Konfiguration des Knotens werden die IP-Adresse und die Subnetzmaske im EEPROM abgelegt.

Hinweis

BootP-Konfiguration wird im EEPROM gespeichert!
Beachten Sie, dass die IP-Adresse und die Subnetzmaske bei der Verwendung von BootP im EEPROM abgelegt werden.

Standardmäßig ist im Feldbuskoppler/-controller das BootP aktiviert.

Bei aktiviertem BootP erwartet der Feldbuskoppler/-controller die permanente Anwesenheit eines BootP-Servers.
Ist jedoch nach einem PowerOn-Reset kein BootP-Server verfügbar, dann bleibt das Netzwerk inaktiv.

Um den Feldbuskoppler/-controller mit der in dem EEPROM hinterlegten IP-Konfiguration zu betreiben, ist nach der Konfiguration von BootP auf die Option „use IP from EEPROM“ umzuschalten.
Dieses erfolgt über das Web-based Management-System auf der entsprechenden feldbuskoppler/-controllerinternen HTML-Seite, die unter dem Link: „Port“ zu erreichen ist.

Beim nächsten Bootvorgang verwendet der Feldbuskoppler/-controller die im EEPROM abgespeicherten Parameter.

Bei einem Fehler in den abgespeicherten Parametern wird über die I/O-LED ein Blinkcode ausgegeben und die Konfiguration über BootP automatisch eingeschaltet.

12.1.2.2 DHCP (Dynamic Host Configuration Protocol)

Die über den Link: „Port“ zu öffnende Feldbuskoppler/-controller-interne HTML-Seite bietet die Option, die Netzwerk-Konfiguration anstatt mit dem BootP-Protokoll auch über die im EEPROM gespeicherten Daten oder über das DHCP durchzuführen.

DHCP (Dynamic Host Configuration Protocol) ist eine Weiterentwicklung von BootP und ist mit diesem rückwärts kompatibel.
Sowohl BootP als auch DHCP weisen dem Feldbusknoten (Client) beim Starten eine IP-Adresse zu, der Ablauf ist dabei der gleiche wie bei BootP.

Bei der Konfiguration der Netzwerkparameter über DHCP sendet der Feldbuskoppler/-controller nach der Initialisierung eigenständig eine Client Anfrage an den DHCP-Server z. B. auf dem angeschlossenen PC.

Über das Protokoll wird eine Broadcast-Anfrage auf Port 67 (DHCP-Server) gesendet, welche die Hardware-Adresse (MAC-ID) des Feldbuskopplers/-controllers enthält.

Der DHCP-Server erhält die Nachricht. Er beinhaltet eine Datenbank, in dem MAC-ID und IP-Adressen einander zugeordnet sind. Wird die MAC-Adresse gefunden, wird eine Broadcast-Antwort über das Netz gesendet.

Der Feldbuskoppler/-controller wartet auf dem vorgegebenen Port 68 auf die Antwort des DHCP-Servers. Ankommende Pakete enthalten unter anderem die IP-Adresse und die MAC-Adresse des Feldbuskopplers/-controllers. An der MAC-Adresse erkennt ein Feldbuskoppler/-controller, ob die Nachricht für ihn bestimmt ist und übernimmt bei Übereinstimmung die gesendete IP-Adresse in sein Netzwerk-Interface.

Erfolgt keine Antwort, so wird die Anfrage nach 4 Sekunden, eine weitere nach 8 Sekunden und nach 16 Sekunden gesendet.
Bleiben alle Anfragen ohne Antwort, so wird ein Blinkcode über die I/O-LED ausgegeben.

Bei der Verwendung von DHCP zur Konfiguration des Knotens werden die IP-Adresse und die Subnetzmaske im EEPROM abgelegt.

Hinweis

DHCP-Konfiguration wird im EEPROM gespeichert!
Beachten Sie, dass die IP-Adresse und Subnetzmaske bei der Verwendung von DHCP im EEPROM abgelegt werden.

Um den Feldbuskoppler/-controller mit der in dem EEPROM hinterlegten IP-Konfiguration zu betreiben, ist nach der Konfiguration über DHCP auf die Option „use IP from EEPROM“ umzuschalten.

Dieses erfolgt über das Web-based Management-System auf der entsprechenden feldbuskoppler/-controllerinternen HTML-Seite, die unter dem Link: „Port“ zu erreichen ist.

Beim nächsten Bootvorgang verwendet der Feldbuskoppler/-controller die im EEPROM abgespeicherten Parameter.

Der Unterschied zwischen BootP und DHCP besteht darin, dass beide verschiedene Zuordnungsverfahren verwenden und die Konfiguration bei DHCP zeitlich begrenzt ist. Der DHCP-Client muss die Konfiguration nach Ablauf der Zeit immer wieder aktualisieren. Im Normalfall werden die gleichen Parameter immer wieder bestätigt vom Server.

BootP ermöglicht die Zuordnung einer festen IP-Adresse für jeden Client, wobei diese Adressen und ihre Reservierung ständig in der BootP-Serverdatenbank gespeichert sind.

DHCP ermöglicht durch diese zeitliche Abhängigkeit die dynamische Zuordnung verfügbarer IP-Adressen durch Client-Leases (Lease-Time, nach der der Client eine neue Adresse anfragt), wobei jede DHCP-Clientadresse temporär in der Serverdatenbank gespeichert ist.

Darüber hinaus ist für DHCP-Clients kein Systemneustart erforderlich, um die Verbindung bzw. Konfiguration mit dem DHCP-Server zu erneuern. Stattdessen gehen die Clients automatisch in bestimmten Zeitabständen einen Neubindungszustand ein, um die Zuordnung der geleasten Adressen am DHCP-Server zu erneuern. Dieser Vorgang wird im Hintergrund ausgeführt und ist für Sie als Anwender transparent.

Es gibt drei verschiedene Betriebsmodi eines DHCP-Servers:

- **Manuelle Zuordnung**
  In diesem Modus werden am DHCP-Server die IP-Adressen bestimmten MAC-Adressen fest zugeordnet. Die Adressen werden der MAC-Adresse auf unbestimmte Zeit zugeteilt.
  Manuelle Zuordnungen werden vor allem dann vorgenommen, wenn der DHCP-Client unter einer festen IP-Adresse erreichbar sein soll.

- **Automatische Zuordnung**
  Bei der automatischen Zuordnung wird am DHCP-Server ein Bereich von IP-Adressen definiert.
  Wenn die Adresse aus diesem Bereich einmal einem DHCP-Client zugeordnet wurde, dann gehört sie diesem auf unbestimmte Zeit, denn auch hier wird die zugewiesene IP-Adresse an die MAC-Adresse gebunden.

- **Dynamische Zuordnung**
  Dieses Verfahren gleicht der automatischen Zuordnung, allerdings hat der DHCP-Server hier in seiner Konfigurationsdatei eine Angabe, wie lange eine bestimmte IP-Adresse an einen Client „vermietet“ werden darf, bevor der Client sich erneut beim Server melden und eine „Verlängerung“ beantragen muss.

Meldet er sich nicht, wird die Adresse frei und kann an einen anderen (oder auch den gleichen) Client neu vergeben werden. Diese vom Administrator bestimmte Zeit heißt Lease-Time (zu Deutsch also: „Mietzeit“).
Manche DHCP-Server vergeben auch von der MAC-Adresse abhängige IP-Adressen, d. h. ein Client bekommt hier selbst nach längerer Netzwerkabstinenz und Ablauf der Lease-Zeit die gleiche IP-Adresse wie zuvor (es sei denn, diese ist inzwischen schon anderweitig vergeben).

Das DHCP dient zum dynamischen Konfigurieren der Netzwerkparameter.
Der ETHERNET TCP/IP-Feldbuscontroller besitzt einen DHCP-Client, der neben der Standard-Option “IP-Adresse” die folgenden Optionen unterstützt:

Tabelle 74: DHCP-Optionen

Option	Bedeutung
[OPT1] Subnetzmaske	32 Bit Adressmaske, die anzeigt, welche Bits der IP-Adresse das Netzwerk und welche die Netzwerkstationen bestimmen.
[OPT2] Zeitzone	Zeitverschiebung zwischen der lokalen Zeit und der UTC (Universal Time Coordinated).
[OPT3] Gateway	IP-Adresse des Routers, der den Zugang zu anderen Netzwerken erlaubt.
[OPT6] DNS-Server	IP-Adresse der Name-Server, die einen Namen in eine IP-Adresse konvertieren. Es können bis zu 2 DNS-Server konfiguriert werden.
[OPT15] Domainename *)	Der Name der Domaine ist die eindeutige Bezeichnung eines Netzwerkes. Der Domainename kann bis zu 32 Zeichen enthalten.
[OPT42] NTP-Server	IP-Adresse des Network Time Servers. Ber der Vergabe eines NTP-Servers wird automatisch der SNTP-Client im Feldbuskoppler/-controller aktiviert.
[OPT51] Lease Time	Hier kann die maximale Dauer definiert werden, wie lange der Feldbuskoppler/-controller die zugewiesene IP-Adresse behält. Die Höchstgrenze der Lease Time beträgt für den Feldbus-koppler/-controller 48 Tage. Dieses ergibt sich aus der internen Timer-Auflösung.
[OPT58] Renewing Time	Die Renewing Time gibt an, ab wann sich der Feldbuskoppler/ -controller um die Erneuerung der Lease-Time kümmern muss. Die Renewing-Time sollte ca. die Hälfte der Lease Time betragen.
[OPT59] Rebinding Time	Die Rebinding Time gibt an, nach welcher Zeit der Feldbuskoppler/-controller seine neue Adresse bekommen haben muss. Die Rebinding Time sollte ca. 7/8 der Lease Time betragen.

*) Im Gegensatz zum BootP unterstützt der DHCP-Client nicht die Vergabe des Hostnamen.

12.1.2.3 HTTP (Hypertext Transfer Protocol)

HTTP ist ein Protokoll, das von WWW (World Wide Web)-Servern zur Weitergabe von Hypermedien, Text, Bildern, Audiodaten usw. verwendet wird. Das HTTP bildet heutzutage die Grundlage des Internets und basiert ebenso wie das BootP-Protokoll auf Anforderungen und Antworten.

Der auf dem Feldbuskoppler/-controller implementierte HTTP-Server dient zum Auslesen der im Feldbuskoppler/-controller abgespeicherten HTML-Seiten. Die

HTML-Seiten geben Auskunft über den Feldbuskoppler/-controller (Zustand, Konfiguration), das Netzwerk und das Prozessabbild.

Auf einigen HTML-Seiten können auch Feldbuskoppler/-controller-Einstellungen über das Web-based Management-System festgelegt und geändert werden, z. B., ob die Netzwerk-Konfiguration des Feldbuskoppler/-controller über das DHCP, das BootP-Protokoll oder aus den gespeicherten Daten im EEPROM erfolgen soll.

Der HTTP-Server benutzt die Portnummer 80.

12.1.2.4 DNS (Domain Name Systems)

Der DNS-Client ermöglicht die Umsetzung von logischen Internet-Namen, wie z. B. www.wago.com in die entsprechende dezimale, mit Trennpunkten dargestellte IP-Adresse über einen DNS-Server. Eine umgekehrte Zuordnung ist ebenso möglich.

Die Adressen der DNS-Server werden mittels DHCP, BootP oder Web-based Management konfiguriert. Es können bis zu zwei DNS-Server angegeben werden. Die Host-Identifikation kann mit zwei Funktionen erfolgen, eine interne Host-Tabelle wird nicht unterstützt.

12.1.2.5 SNTP-Client (Simple Network Time Protocol)

Der SNTP-Client wird für die Synchronisation der Uhrzeit zwischen einem Time-Server (NTP- und SNTP-Server der Version 3 und 4) und der Systemzeit im Feldbuskoppler/-controller verwendet. Das Protokoll wird über einen UDP-Port abgearbeitet. Es wird ausschließlich die Unicast-Adressierung unterstützt.

Konfiguration des SNTP-Client

Die Konfiguration des SNTP-Client wird über das Web-based Management vorgenommen. Folgende Parameter müssen eingestellt werden:

Tabelle 75: Bedeutung der SNTP-Parameter

Parameter	Bedeutung
WBM-Seite „TCP/IP“ → „(S)NTP Server“	Die Adressvergabe kann über eine IP- Adresse vorgenommen werden.
WBM-Seite „TCP/IP“ → „SNTP Update Time (sec, max. 65535)“	Die Update-Time gibt das Intervall in Sekunden an, in der die Synchronisierung mit dem Time-Server erfolgen soll.
WBM-Seite „Clock“ → „Timezone (+/- hour:minute)”	Für den Betrieb der ETHERNET Feldbuskoppler/-controller mit SNTP in verschiedenen Ländern muss eine Zeitzone angegeben werden. Die Einstellung der Zeitzone bezieht sich relativ zur GMT (Greenwich-Mean-Time). Es kann ein Bereich von -12 bis +14 Stunden angegeben werden.
WBM-Seite „Port“ → „SNTP”	Gibt an, ob der SNTP-Client aktiviert oder deaktiviert werden soll.

12.1.2.6 FTP-Server (File Transfer Protocol)

Das File Transfer Protokoll ermöglicht es, Dateien unabhängig vom Aufbau des Betriebssystems zwischen verschiedenen Netzwerkteilnehmern auszutauschen. Bei dem ETHERNET Feldbuskoppler/-controller dient FTP dazu, die vom Anwender erstellten HTML-Seiten, das IEC-61131-Programm und den IEC-61131-Source-Code in dem (programmierbaren) Feldbuskoppler/-controller abzuspeichern und auszulesen.

Für das Dateisystem steht ein Gesamtspeicher von 2 MB zur Verfügung.

Hinweis

Zyklen für Flash auf 1 Million begrenzt!
Bis zu 1 Million Schreibzyklen pro Sektor sind beim Beschreiben des Flash für das Dateisystem möglich. Das Dateisystem unterstützt „Wear-Leveling“, damit nicht immer auf dieselben Sektoren geschrieben wird.

Information

Weitere Information zu den implementierten Protokollen
Die in dem Feldbuskoppler/-controller jeweils implementierten und unterstützten Protokolle sind in dem Kapitel „Technische Daten“ zu dem Feldbuskoppler/-controller aufgelistet.

12.1.3 Anwendungsprotokolle

Über die implementierten Anwendungsprotokolle ist mit dem Feldbuskoppler/-controller die entsprechende feldbusspezifische Kommunikation möglich. Dadurch hat der Anwender einen einfachen Zugriff von dem jeweiligen Feldbus auf den Feldbusknoten.

Die in dem Feldbuskoppler/-controller implementierten feldbusspezifischen Anwendungsprotokolle sind im Einzelnen in den nachfolgenden Kapiteln ausführlich beschrieben.

12.2 MODBUS-Funktionen

12.2.1 Allgemeines

MODBUS ist ein herstellerunabhängiger, offener Feldbusstandard für vielfältige Anwendungen in der Fertigungs- und Prozessautomation.

Das MODBUS-Protokoll ist nach dem aktuellen Internet-Draft der IETF (Internet Engineering Task Force) implementiert und erfüllt folgende Funktionen:

- Übermitteln des Prozessabbildes
- Übermitteln der Feldbusvariablen
- Übermitteln verschiedener Einstellungen und Informationen des Kopplers/Controllers über den Feldbus

Der Datentransport in der Feldebene erfolgt über TCP sowie über UDP.

Das MODBUS/TCP-Protokoll ist eine Variante des MODBUS-Protokolls, das für die Kommunikation über TCP/IP-Verbindungen optimiert wurde.

Alle Datenpakete werden über eine TCP-Verbindung mit der Portnummer 502 gesendet.

MODBUS/TCP-Datenpaket

Der allgemeine MODBUS/TCP-Header stellt sich folgendermaßen dar:

Tabelle 76: MODBUS/TCP-Header

Byte	0	1	2	3	4	5	6	7	8...n
	Kennung (wird vom Empfänger eingetragen)		Protokollkennung (immer 0 für MODBUS/TCP)		Feldlänge (Highbyte, Lowbyte)		Einheiten-kennung (Slave-Adresse)	MODBUS-Funktionscode	Daten

Information

Weitere Information

Der Telegrammaufbau ist spezifisch für die einzelnen Funktionen und deshalb detailliert in den Beschreibungen der MODBUS-Funktionscodes erläutert.

Für das MODBUS-Protokoll werden 15 Verbindungen über TCP zur Verfügung gestellt. Damit ist es möglich, von 15 Stationen zeitgleich digitale und analoge Ausgangsdaten an einem Feldbusknoten direkt auszulesen und spezielle Funktionen durch einfache MODBUS-Funktionscodes auszuführen.

Zu diesem Zweck sind eine Reihe von MODBUS-Funktionen aus der „Open MODBUS/TCP Specification“ realisiert.

Information

Weitere Information

Weiterführende Informationen zu der „Open MODBUS/TCP Specification“ finden Sie im Internet unter: http://www.modbus.org

Das MODBUS-Protokoll basiert dabei im Wesentlichen auf den folgenden Grunddatentypen:

Tabelle 77: Grunddatentypen des MODBUS-Protokolls

Datentyp	Länge	Beschreibung
Discrete Inputs	1 Bit	Digitale Eingänge
Coils	1 Bit	Digitale Ausgänge
Input Register	16 Bit	Analoge Eingänge
Holding Register	16 Bit	Analoge Ausgänge

Für jeden Grunddatentyp sind ein oder mehr Funktionscodes definiert.

Mit diesen Funktionen können gewünschte binäre oder analoge Ein- und Ausgangsdaten und interne Variablen aus dem Feldbusknoten gesetzt oder direkt ausgelesen werden.

Tabelle 78: Auflistung der in dem Feldbuscontroller realisierten MODBUS-Funktionen

Funktionscode		Funktionsname	Zugriffsart und -beschreibung	Zugriff auf Ressourcen
FC1	0x01	Read Coils	Lesen mehrerer Ausgangsbits	**R:** Prozessabbild, PFC-Variablen
FC2	0x02	Read Discrete Inputs	Lesen mehrerer Eingangsbits	**R:** Prozessabbild, PFC-Variablen
FC3	0x03	Read Holding Registers	Lesen mehrerer Eingangsregister	**R:** Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC4	0x04	Read Input Registers	Lesen mehrerer Eingangsregister	**R:** Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC5	0x05	Write Single Coil	Schreiben eines einzelnen Ausgangsbits	**W:** Prozessabbild, PFC-Variablen
FC6	0x06	Write Single Register	Schreiben eines einzelnen Ausgangsregisters	**W:** Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC11	0x0B	Get Comm Event Counters	Kommunikationsereigniszähler	**R:** Keine
FC15	0x0F	Write Multiple Coils	Schreiben mehrerer Ausgangsbits	**W:** Prozessabbild, PFC-Variablen
FC16	0x10	Write Multiple Registers	Schreiben mehrerer Ausgangsregister	**W:** Prozessabbild, PFC-Variablen, Interne Variablen, NOVRAM
FC22	0x16	Mask Write Register	Schreiben einzelner Bits eines einzelnen Ausgangsregisters durch Maskierung	**W:** Prozessabbild, PFC-Variablen, NOVRAM

FC23 0x17	Read/Write Multiple Registers	Lesen und Schreiben mehrerer Ausgangsregister	**R/W:** Prozessabbild, PFC-Variablen, NOVRAM

Um eine gewünschte Funktion auszuführen, wird der entsprechende Funktionscode und die Adresse des ausgewählten Ein- oder Ausgangskanals angegeben.

Hinweis

Bei der Adressierung auf das verwendete Zahlensystem achten!
Die aufgeführten Beispiele verwenden als Zahlenformat das Hexadezimalsystem (Bsp.: 0x000). Die Adressierung beginnt mit 0.
Je nach Software und Steuerung kann das Format und der Beginn der Adressierung variieren. Alle Adressen sind in diesem Fall dementsprechend umzurechnen.

12.2.2 Anwendung der MODBUS-Funktionen

Die grafische Übersicht zeigt anhand eines exemplarischen Feldbusknotens den Zugriff einiger MODBUS-Funktionen auf die Daten des Prozessabbildes.

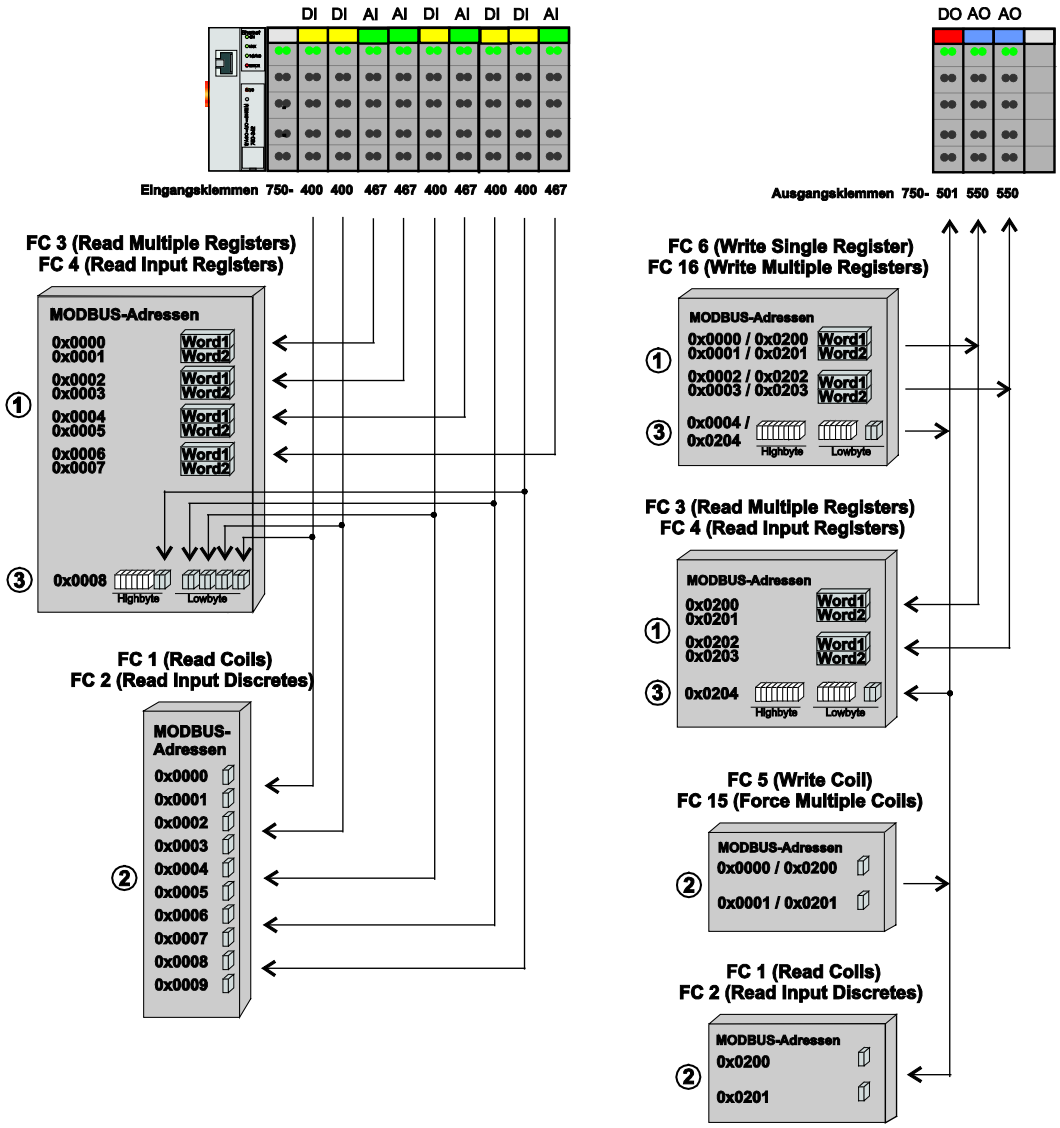


Abbildung 72: Anwendung von MODBUS-Funktionen für einen Feldbuskoppler/-controller

Hinweis

Registerfunktionen für analoge Signale, Coil-Funktionen für binäre Signale verwenden!

Es ist sinnvoll, auf die analogen Signale mit Registerfunktionen ① und auf die binären Signale mit Coil-Funktionen ② zuzugreifen. Wird auf die binären Signale lesend oder schreibend mit Registerfunktionen ③ zugegriffen, verschieben sich die Adressen, sobald weitere analoge Busklemmen an dem Feldbuskoppler/-controller betrieben werden.

12.2.3 Beschreibung der MODBUS-Funktionen

Alle implementierten MODBUS-Funktionen werden in der folgenden Weise ausgeführt:

1. Mit der Eingabe eines Funktionscodes stellt der MODBUS/TCP-Master (z. B. ein PC) eine entsprechende Anfrage (Request) an den WAGO-Feldbusknoten.

2. Der WAGO-Feldbusknoten sendet ein Telegramm als Antwort (Response) an den Master zurück.

Empfängt der WAGO-Feldbusknoten eine fehlerhafte Anfrage, sendet dieser ein Fehlertelegramm (Exception) an den Master zurück. Dabei hat der im Fehlertelegramm befindliche Exception-Code die folgende Bedeutung:

Tabelle 79: Exception-Codes

Exception-Code	Bedeutung
0x01	Illegal function
0x02	Illegal data address
0x03	Illegal data value
0x04	Slave device failure

In den folgenden Kapiteln wird für jeden Funktionscode der Telegrammaufbau von Request, Response und Exception mit Beispielen beschrieben.

Hinweis

Lesen und Schreiben der Ausgänge bei FC1 bis FC4 auch durch Hinzuaddieren eines Offsets möglich!
Bei den Lesefunktionen (FC1 ... FC4) können Sie zusätzlich die Ausgänge schreiben und zurücklesen, indem Sie für Adressen in dem Bereich [0_{hex} ... FF_{hex}] ein Offset von 200_{hex} (0x0200) und für Adressen in dem Bereich [6000_{hex} ... $62FC_{hex}$] ein Offset von 1000_{hex} (0x1000) zu der MODBUS-Adresse hinzu addieren.

12.2.3.1 Funktionscode FC1 (Read Coils)

Diese Funktion liest den Inhalt mehrerer Eingangs- und Ausgangsbits.

Aufbau des Request

Die Anfrage bestimmt die Startadresse und die Anzahl der zu lesenden Bits. Beispiel: Eine Anfrage, mit welcher Bit 0 bis Bit 7 gelesen werden.

Tabelle 80: Aufbau des Request für den Funktionscode FC1

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x01
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0008

Aufbau der Response

Die aktuellen Werte der abgefragten Bits werden in das Datenfeld geschrieben. Eine 1 entspricht dabei dem Zustand ON und eine 0 dem Zustand OFF. Das niederwertigste Bit des ersten Datenbytes enthält das erste Bit der Anfrage. Die anderen Bits folgen aufsteigend. Falls die Anzahl der Eingänge kein Vielfaches von 8 ist, werden die verbleibenden Bits des letzten Datenbytes mit Nullen aufgefüllt.

Tabelle 81: Aufbau der Response für den Funktionscode FC1

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x01
Byte 8	Byte count	0x01
Byte 9	Bit values	0x12

Der Status der Eingänge 7 bis 0 wird als Byte-Wert 0x12 oder Binärwert 0001 0010 angezeigt. Eingang 7 ist das Bit mit dem höchsten Wert, Eingang 0 ist das Bit mit dem niedrigsten Wert dieses Bytes. Die Zuordnung erfolgt von 7 bis 0 wie folgt:

Tabelle 82: Zuordnung der Eingänge

	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF
Bit	0	0	0	1	0	0	1	0
Coil	7	6	5	4	3	2	1	0

Aufbau der Exception

Tabelle 83: Aufbau der Exception für den Funktionscode FC1

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x81
Byte 8	Exception code	0x01 oder 0x02

12.2.3.2 Funktionscode FC2 (Read Discrete Inputs)

Diese Funktion liest den Inhalt mehrerer Eingangsbits (digitale Eingänge).

Aufbau des Request

Die Anfrage bestimmt die Startadresse und die Anzahl der zu lesenden Bits. Beispiel: Eine Anfrage, mit welcher Bit 0 bis Bit 7 gelesen werden.

Tabelle 84: Aufbau des Request für den Funktionscode FC2

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x02
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0008

Aufbau der Response

Die aktuellen Werte der abgefragten Bits werden in das Datenfeld geschrieben. Eine 1 entspricht dabei dem Zustand ON und eine 0 dem Zustand OFF. Das niederwertigste Bit des ersten Datenbytes enthält das erste Bit der Anfrage. Die anderen Bits folgen aufsteigend. Falls die Anzahl der Eingänge kein Vielfaches von 8 ist, werden die verbleibenden Bits des letzten Datenbytes mit Nullen aufgefüllt.

Tabelle 85: Aufbau der Response für den Funktionscode FC2

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x02
Byte 8	Byte count	0x01
Byte 9	Bit values	0x12

Der Status der Eingänge 7 bis 0 wird als Byte-Wert 0x12 oder Binärwert 0001 0010 angezeigt. Eingang 7 ist das Bit mit dem höchsten Wert, Eingang 0 ist das Bit mit dem niedrigsten Wert dieses Bytes. Die Zuordnung erfolgt von 7 bis 0 wie folgt:

Tabelle 86: Zuordnung der Eingänge

	OFF	OFF	OFF	ON	OFF	OFF	ON	OFF
Bit	0	0	0	1	0	0	1	0
Coil	7	6	5	4	3	2	1	0

Aufbau der Exception

Tabelle 87: Aufbau der Exception für den Funktionscode FC2

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x82
Byte 8	Exception code	0x01 oder 0x02

12.2.3.3 Funktionscode FC3 (Read Holding Registers)

Diese Funktion dient dazu, eine Anzahl von Eingangsworten (Eingangsregister) zu lesen.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Startwortes (Startregister) und die Anzahl der Register, die gelesen werden. Die Adressierung beginnt mit 0.
Beispiel: Abfrage der Register 0 und 1.

Tabelle 88: Aufbau des Request für den Funktionscode FC3

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x03
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Response

Die Registerdaten der Antwort werden als 2 Bytes pro Register gepackt. Das erste Byte enthält dabei die höherwertigen Bits, das zweite Byte die niederwertigen.

Tabelle 89: Aufbau der Response für den Funktionscode FC3

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x03
Byte 8	Byte count	0x04
Byte 9, 10	Value register 0	0x1234
Byte 11, 12	Value register 1	0x2345

Aus der Antwort ergibt sich, dass Register 0 den Wert 0x1234 und Register 1 den Wert 0x2345 enthält.

Aufbau der Exception

Tabelle 90: Aufbau der Exception für den Funktionscode FC3

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x83
Byte 8	Exception code	0x01 oder 0x02

12.2.3.4 Funktionscode FC4 (Read Input Registers)

Diese Funktion dient dazu, eine Anzahl von Eingangsworten (Eingangsregister) zu lesen.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Startwortes (Startregister) und die Anzahl der Register, die gelesen werden sollen. Die Adressierung beginnt mit 0.
Beispiel: Abfrage der Register 0 und 1.

Tabelle 91: Aufbau des Request für den Funktionscode FC4

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x04
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Response

Die Registerdaten der Antwort werden als 2 Bytes pro Register gepackt. Das erste Byte enthält dabei die höherwertigen Bits, das zweite die niederwertigen.

Tabelle 92: Aufbau der Response für den Funktionscode FC4

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x04
Byte 8	Byte count	0x04
Byte 9, 10	Value register 0	0x1234
Byte 11, 12	Value register 1	0x2345

Aus der Antwort ergibt sich, dass Register 0 den Wert 0x1234 und Register 1 den Wert 0x2345 enthält.

Aufbau der Exception

Tabelle 93: Aufbau der Exception für den Funktionscode FC4

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x84
Byte 8	Exception code	0x01 oder 0x02

12.2.3.5 Funktionscode FC5 (Write Single Coil)

Diese Funktion dient dazu, ein digitales Ausgangsbit zu schreiben.

Aufbau des Request

Die Anfrage bestimmt die Adresse des Ausgangsbits. Die Adressierung beginnt mit 0.
Beispiel: Setzen des 2. Ausgangsbits (Adresse 1).

Tabelle 94: Aufbau des Request für den Funktionscode FC5

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x05
Byte 8, 9	Reference number	0x0001
Byte 10	ON/OFF	0xFF
Byte 11		0x00

Aufbau der Response

Tabelle 95: Aufbau der Response für den Funktionscode FC5

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x05
Byte 8, 9	Reference number	0x0001
Byte 10	Value	0xFF
Byte 11		0x00

Aufbau der Exception

Tabelle 96: Aufbau der Exception für den Funktionscode FC5

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01, 0x02 oder 0x03

12.2.3.6 Funktionscode FC6 (Write Single Register)

Diese Funktion schreibt einen Wert in ein einzelnes Ausgangswort (Ausgangsregister).

Aufbau des Request

Die Adressierung beginnt mit 0. Die Anfrage bestimmt die Adresse des Ausgangswortes, das gesetzt werden soll. Der zu setzende Wert wird im Anfragedatenfeld bestimmt.
Beispiel: Setzen des zweiten Ausgangskanal auf den Wert 0x1234.

Tabelle 97: Aufbau des Request für den Funktionscode FC6

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0006
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x06
Byte 8, 9	Reference number	0x0001
Byte 10, 11	Register value	0x1234

Aufbau der Response

Die Antwort ist ein Echo der Anfrage.

Tabelle 98: Aufbau der Response für den Funktionscode FC6

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x06
Byte 8, 9	Reference number	0x0001
Byte 10, 11	Register value	0x1234

Aufbau der Exception

Tabelle 99: Aufbau der Exception für den Funktionscode FC6

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.2.3.7 Funktionscode FC11 (Get Comm Event Counter)

Diese Funktion gibt ein Statuswort und einen Ereigniszähler aus dem Kommunikationsereigniszähler des Feldbuscontrollers zurück. Die übergeordnete Steuerung kann mit diesem Zähler feststellen, ob der Feldbuscontroller die Nachrichten fehlerlos verarbeitet hat.

Nach jeder erfolgreichen Nachrichtenverarbeitung wird der Zähler hochgezählt. Fehlermeldungen oder Zählerabfragen werden nicht mitgezählt.

Aufbau des Request

Tabelle 100: Aufbau des Request für den Funktionscode FC11

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0002
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x0B

Aufbau der Response

Die Antwort enthält ein 2-Byte-Statuswort und einen 2-Byte-Ereigniszähler. Das Statuswort besteht aus Nullen.

Tabelle 101: Aufbau der Response für den Funktionscode FC11

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x0B
Byte 8, 9	Status	0x0000
Byte 10, 11	Event count	0x0003

Der Ereigniszähler zeigt, dass 3 (0x0003) Ereignisse gezählt wurden.

Aufbau der Exception

Tabelle 102: Aufbau der Exception für den Funktionscode FC11

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.2.3.8 Funktionscode FC15 (Write Multiple Coils)

Mit dieser Funktion wird eine Anzahl von bis zu 256 Ausgangsbits auf 1 oder 0 gesetzt.

Aufbau des Request

Das erste Bit wird mit 0 adressiert. In der Anfrage werden die Bits spezifiziert, die gesetzt werden sollen. Die geforderten 1-oder 0-Zustände werden durch die Inhalte des Anfragedatenfeldes bestimmt.

In diesem Beispiel werden 16 Bits beginnend mit Adresse 0 gesetzt. Die Anfrage enthält 2 Bytes mit dem Wert 0xA5F0 also 1010 0101 1111 0000 binär.

Das erste Byte überträgt den Wert 0xA5 an die Adresse 7 bis 0, wobei Bit 0 das niederwertigste Bit ist. Das nächste Byte überträgt den Wert 0xF0 an die Adresse 15 bis 8, wobei Bit 8 das niederwertigste Bit ist.

Tabelle 103: Aufbau des Request für den Funktionscode FC15

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0009
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x0F
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0010
Byte 12	Byte count	0x02
Byte 13	Data byte1	0xA5
Byte 14	Data byte2	0xF0

Aufbau der Response

Tabelle 104: Aufbau der Response für den Funktionscode FC15

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x0F
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Bit count	0x0010

Aufbau der Exception

Tabelle 105: Aufbau der Exception für den Funktionscode FC15

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x8F
Byte 8	Exception code	0x01 oder 0x02

12.2.3.9 Funktionscode FC16 (Write Multiple Registers)

Diese Funktion schreibt Werte in eine Anzahl von Ausgangsworten (Ausgangsregister).

Aufbau des Request

Das erste Register wird mit 0 adressiert.
Die Anfragenachricht bestimmt die Register, die gesetzt werden sollen.
Pro Register werden 2 Byte an Daten gesendet.
Beispiel: Die Daten in den beiden Registern 0 und 1 werden gesetzt.

Tabelle 106: Aufbau des Request für den Funktionscode FC16

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x000B
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002
Byte 12	Byte count	0x04
Byte 13, 14	Register value 1	0x1234
Byte 15, 16	Register value 2	0x2345

Aufbau der Response

Tabelle 107: Aufbau der Response für den Funktionscode FC16

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	Word count	0x0002

Aufbau der Exception

Tabelle 108: Aufbau der Exception für den Funktionscode FC16

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.2.3.10 Funktionscode FC22 (Mask Write Register)

Diese Funktion dient dazu einzelne Bits innerhalb eines Registers zu manipulieren.

Aufbau des Request

Tabelle 109: Aufbau des Request für den Funktionscode FC22

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x0002
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x16
Byte 8, 9	Reference number	0x0000
Byte 10, 11	AND mask	0x0000
Byte 12, 13	OR mask	0xAAAA

Aufbau der Response

Tabelle 110: Aufbau der Response für den Funktionscode FC22

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x10
Byte 8, 9	Reference number	0x0000
Byte 10, 11	AND mask	0x0000
Byte 12, 13	OR mask	0xAAAA

Aufbau der Exception

Tabelle 111: Aufbau der Exception für den Funktionscode FC22

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x85
Byte 8	Exception code	0x01 oder 0x02

12.2.3.11 Funktionscode FC23 (Read/Write Multiple Registers)

Diese Funktion liest Registerwerte aus und schreibt Werte in eine Anzahl von Ausgangsworten (Ausgangsregister). Der Schreibzugriff wird vor dem Lesezugriff ausgeführt.

Aufbau des Request

Das erste Register wird mit 0 adressiert.
Die Anfragenachricht bestimmt die Register, die gelesen und gesetzt werden sollen.
Pro Register werden 2 Byte an Daten gesendet.
Beispiel: Die Daten in dem Register 3 werden auf den Wert 0x0123 gesetzt. Aus den beiden Registern 0 und 1 werden die Werte 0x0004 und 0x5678 gelesen.

Tabelle 112: Aufbau des Request für den Funktionscode FC23

Byte	Feldname	Beispiel
Byte 0, 1	Transaction identifier	0x0000
Byte 2, 3	Protocol identifier	0x0000
Byte 4, 5	Length field	0x000D
Byte 6	Unit identifier	0x01 nicht verwendet
Byte 7	MODBUS function code	0x17
Byte 8, 9	Reference number for read	0x0000
Byte 10, 11	Word count for read (1-125)	0x0002
Byte 12, 13	Reference number for write	0x0003
Byte 14, 15	Word count for write (1-100)	0x0001
Byte 16	Byte count (2 x word count for write)	0x02
Byte 17...(B+16)	Register values (B = Byte count)	0x0123

Aufbau der Response

Tabelle 113: Aufbau der Response für den Funktionscode FC23

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x17
Byte 8	Byte count (2 x word count for read)	0x04
Byte 9...(B+1)	Register values (B = Byte count)	0x0004 oder 0x5678

Aufbau der Exception

Tabelle 114: Aufbau der Exception für den Funktionscode FC23

Byte	Feldname	Beispiel
...		
Byte 7	MODBUS function code	0x97
Byte 8	Exception code	0x01 oder 0x02

Hinweis

Ergebnisse in überlappenden Registerbereichen sind undefiniert!
Wenn sich für das Lesen und Schreiben Registerbereiche überlappen, sind die Ergebnisse undefiniert.

12.2.4 MODBUS-Register-Mapping

In den folgenden Tabellen werden die MODBUS-Adressierung und die entsprechende IEC-61131-Adressierung für das Prozessabbild, die PFC-Variablen, die NOVRAM-Daten und die internen Variablen dargestellt.

Über die Registerdienste lassen sich die Zustände von komplexen und digitalen Busklemmen ermitteln oder verändern.

Registerzugriff Lesen (mit FC3, FC4 und FC23)

Tabelle 115: Registerzugriff Lesen (mit FC3, FC4 und FC23)

MODBUS-Adresse [dez]	MODBUS-Adresse [hex]	IEC-61131-Adresse	Speicherbereich
0...255	0x0000...0x00FF	%IW0...%IW255	Physical-Input-Area (1) First 256 Words of physical input data
256...511	0x0100...0x01FF	%QW256...%QW511	PFC-OUT-Area Flüchtige SPS-Ausgangsvariablen
512...767	0x0200...0x02FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
768...1023	0x0300...0x03FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...12287	0x1000...0x2FFF	-	Konfigurationsregister (siehe Kapitel „Konfigurationsregister“)
12288...20479	0x3000...0x4FFF	%MW0...%MW4095	NOVRAM 8 kB retain memory *) *) In Zielsystemeinstellungen RETAIN auf 0, Merker auf MAX (8 kB)
24576...25339	0x6000...0x62FB	%IW512...%IW1275	Physical-Input-Area (2) Additional 764 Words physical input data
25340...28671	0x62FC...0x6FFF	-	MODBUS-Exception: “Illegal data address”
28672...29435	0x7000...0x72FB	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
29436...65535	0x72FC...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Registerzugriff Schreiben (mit FC6, FC16, FC22 und FC23)

Tabelle 116: Registerzugriff Schreiben (mit FC6, FC16, FC22 und FC23)

MODBUS-Adresse		IEC-61131-Adresse	Speicherbereich
[dez]	[hex]		
0...255	0x0000...0x00FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
256...511	0x0100...0x01FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
512...767	0x0200...0x02FF	%QW0...%QW255	Physical-Output-Area (1) First 256 Words of physical output data
768...1023	0x0300...0x03FF	%IW256...%IW511	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...12287	0x1000...0x2FFF	-	Konfigurationsregister (siehe Kapitel „Konfigurationsregister“)
12288...20479	0x3000...0x4FFF	%MW0...%MW4095	NOVRAM 8 kB retain memory *) *) In Zielsystemeinstellungen RETAIN auf 0, Merker auf MAX (8 kB)
24576...25339	0x6000...0x62FB	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
25340...28671	0x62FC...0x6FFF	-	MODBUS-Exception: “Illegal data address”
28672...29435	0x7000...0x72FB	%QW512...%QW1275	Physical-Output-Area (2) Additional 764 Words physical output data
29436...65535	0x72FC...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Die digitalen MODBUS-Dienste (Coil-Dienste) sind Bitzugriffe, mit denen sich die Zustände von digitalen Busklemmen ermitteln oder verändern lassen. Komplexe Busklemmen sind mit diesen Diensten nicht erreichbar und werden ignoriert. Deshalb wird bei der Adressierung der digitalen Kanäle wieder mit 0 begonnen, so dass die MODBUS-Adresse immer identisch mit der Kanalnummer ist (der 47. digitale Eingang hat beispielsweise die MODBUS-Adresse „46“).

Bitzugriff Lesen (mit FC1 und FC2)

Tabelle 117: Bitzugriff Lesen (mit FC1 und FC2)

MODBUS-Adresse		Speicherbereich	Beschreibung
[dez]	[hex]		
0...511	0x0000...0x01FF	Physical Input Area (1)	First 512 digital inputs
512...1023	0x0200...0x03FF	Physical Output Area (1)	First 512 digital outputs
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...8191	0x1000...0x1FFF	%QX256.0...%QX511.15	PFC-OUT-Area Flüchtige SPS-Ausgangsvariablen
8192...12287	0x2000...0x2FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
12288...32767	0x3000...0x7FFF	%MX0...%MX1279.15	NOVRAM 2 kB retain memory
32768...34295	0x8000...0x85F7	Physical Input Area (2)	Starts with the 513th and ends with the 2039 th digital input
34296...36863	0x85F8...0x8FFF	-	MODBUS-Exception: “Illegal data address”
36864...38391	0x9000...0x95F7	Physical Output Area (2)	Starts with the 513 th and ends with the 2039 th digital output
38392...65535	0x95F8...0xFFFF	-	MODBUS-Exception: “Illegal data address”

Bitzugriff Schreiben (mit FC5 und FC15)

Tabelle 118: Bitzugriff Schreiben (mit FC5 und FC15)

MODBUS-Adresse		Speicherbereich	Beschreibung
[dez]	[hex]		
0...511	0x0000...0x01FF	Physical-Output-Area (1)	First 512 digital outputs
512...1023	0x0200...0x03FF	Physical-Output-Area (1)	First 512 digital outputs
1024...4095	0x0400...0x0FFF	-	MODBUS-Exception: “Illegal data address”
4096...8191	0x1000...0x1FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
8192...12287	0x2000...0x2FFF	%IX256.0...%IX511.15	PFC-IN-Area Flüchtige SPS-Eingangsvariablen
12288...32767	0x3000...0x7FFF	%MX0...%MX1279.15	NOVRAM 2 kB retain memory
32768...34295	0x8000...0x85F7	Physical-Output-Area (2)	Starts with the 513th and ends with the 2039 th digital input
34296...36863	0x85F8...0x8FFF	-	MODBUS-Exception: “Illegal data address”
36864...38391	0x9000...0x95F7	Physical-Output-Area (2)	Starts with the 513 th and ends with the 2039 th digital output
38392...65535	0x95F8...0xFFFF	-	MODBUS-Exception: “Illegal data address”

12.2.5 MODBUS-Register

Tabelle 119: MODBUS-Register

Register-adresse	Zugriff	Länge (Wort)	Beschreibung
0x1000	R/W	1	Watchdog-Zeit lesen/schreiben
0x1001	R/W	1	Watchdog-Codiermaske 1...16
0x1002	R/W	1	Watchdog-Codiermaske 17...32
0x1003	R/W	1	Watchdog-Trigger
0x1004	R	1	Minimale Triggerzeit
0x1005	R/W	1	Watchdog stoppen (Schreibsequenz 0xAAAA, 0x5555)
0x1006	R	1	Watchdog-Status
0x1007	R/W	1	Watchdog neu starten (Schreibsequenz 0x1)
0x1008	R/W	1	Watchdog stoppen (Schreibsequenz 0x55AA oder 0xAA55)
0x1009	R/W	1	MODBUS und HTTP schließen bei Watchdog Time-out
0x100A	R/W	1	Watchdog-Konfiguration
0x100B	W	1	Watchdog-Parameter speichern (Schreibsequenz 0x55AA oder 0xAA55)
0x1020	R	1… 2	LED Error-Code
0x1021	R	1	LED Error-Argument
0x1022	R	1… 4	Anzahl analoger Ausgangsdaten im Prozessabbild (in Bits)
0x1023	R	1… 3	Anzahl analoger Eingangsdaten im Prozessabbild (in Bits)
0x1024	R	1… 2	Anzahl digitaler Ausgangsdaten im Prozessabbild (in Bits)
0x1025	R	1	Anzahl digitaler Eingangsdaten im Prozessabbild (in Bits)
0x1028	R/W	1	Boot-Konfiguration
0x1029	R/W	1 … 9	MODBUS/TCP-Statistik (Schreibsequenz 0x55AA oder 0xAA55)
0x102A	R	1	Anzahl der TCP-Verbindungen
0x102B	W	1	KBUS-Reset
0x1030	R/W	1	Konfiguration MODBUS/TCP-Time-out
0x1031	R	1 … 3	Lesen der MAC-ID des Feldbuskopplers/-controllers
0x1035	R/W	1	Timeoffset RTC
0x1036	R/W	1	Daylight Saving
0x1037	R/W	1	Modbus Response Delay (ms)
0x1050	R	3	Diagnose angeschlossener Busklemmen
0x2000	R	1 … 9	Konstante 0x0000
0x2001	R	1 … 8	Konstante 0xFFFF
0x2002	R	1 … 7	Konstante 0x1234
0x2003	R	1 … 6	Konstante 0xAAAA
0x2004	R	1 … 5	Konstante 0x5555
0x2005	R	1 … 4	Konstante 0x7FFF
0x2006	R	1 … 3	Konstante 0x8000
0x2007	R	1 … 2	Konstante 0x3FFF
0x2008	R	1	Konstante 0x4000
0x2010	R	1	Firmware-Version
0x2011	R	1	Seriencode
0x2012	R	1	Feldbuskoppler/-controller-Code
0x2013	R	1	Firmware-Versionen Major-Revision
0x2014	R	1	Firmware-Versionen Minor-Revision

Tabelle 120: MODBUS-Register (Fortsetzung)

Register-adresse	Zugriff	Länge (Wort)	Beschreibung
0x2020	R	1 … 16	Kurzbeschreibung Feldbuskoppler/-controller
0x2021	R	1 … 8	Kompilierzeit der Firmware
0x2022	R	1 … 8	Kompilierdatum der Firmware
0x2023	R	1 … 32	Angabe des Firmware-Loaders
0x2030	R	1 … 65	Beschreibung der angeschlossenen Busklemmen (Busklemme 0...64)
0x2031	R	1 … 64	Beschreibung der angeschlossenen Busklemmen (Busklemme 65...129)
0x2032	R	1 … 64	Beschreibung der angeschlossenen Busklemmen (Busklemme 130...194)
0x2033	R	1 … 63	Beschreibung der angeschlossenen Busklemmen (Busklemme 195...255)
0x2040	W	1	Software-Reset (Schreibsequenz 0x55AA oder 0xAA55)
0x2041	W	1	Format Flash-Disk
0x2042	W	1	HTML-Seiten aus der Firmware extrahieren
0x2043	W	1	Werkseinstellungen

12.2.5.1 Zugriff auf Registerwerte

Um lesend oder schreibend auf Registerwerte zugreifen zu können, verwenden Sie eine beliebige MODBUS-Anwendung. Neben kommerziellen Anwendungen (beispielsweise „ModScan“) stehen Ihnen auch kostenfreie Programme zur Verfügung (siehe Internetseite http://www.modbus.org/tech.php).

Die nachfolgenden Kapitel beschreiben den Zugriff auf die Register und ihre Werte.

12.2.5.2 Watchdog-Register

Der Watchdog überwacht die Datenübertragung zwischen übergeordneter Steuerung und Feldbuskoppler/-controller. Dazu wird von der übergeordneten Steuerung eine Zeitfunktion (Time-out) in dem Feldbuscontroller zyklisch angestoßen.

Bei fehlerfreier Kommunikation kann diese Zeit ihren Endwert nicht erreichen, weil sie zuvor immer wieder neu gestartet wird. Läuft die Zeit jedoch ohne Unterbrechung ab, liegt ein Feldbusausfall vor.

In diesem Fall antwortet der Feldbuskoppler/-controller auf alle folgenden MODBUS-TCP/IP-Anfragen mit dem Exception-Code 0x0004 (Slave Device Failure).

Im Feldbuskoppler/-controller sind gesonderte Register für die Ansteuerung und für die Statusabfrage des Watchdogs durch die übergeordnete Steuerung vorhanden (Registeradressen 0x1000 bis 0x1008).

Nach dem Einschalten der Versorgungsspannung ist der Watchdog noch nicht aktiviert. Zunächst ist der Time-out-Wert festzulegen (Register 0x1000). Der Watchdog kann dann zum ersten Mal aktiviert werden, indem im Masken-

Register (0x1001) ein Funktionscode geschrieben wird, der ungleich 0 ist. Die Möglichkeit zur anschließenden Aktivierung nach einem Time-out besteht darin, in das Toggle-Register (0x1003) oder in das Register 0x1007 einen von 0 abweichenden Wert zu schreiben.

Durch das Lesen der minimalen Triggerzeit (Register 0x1004) wird festgestellt, ob die Watchdog-Fehlerreaktion aktiviert wurde. Falls dieser Zeitwert 0 ist, wird ein Feldbusausfall angenommen. Der Watchdog kann dann entsprechend der zuvor genannten beiden Möglichkeiten (mittels Register 0x1003 oder 0x1007) neu gestartet werden.

Wenn der Watchdog einmal gestartet wurde, kann er vom Anwender aus Sicherheitsgründen lediglich über einen bestimmten Weg gestoppt werden (Register 0x1005 oder 0x1008).

Die Watchdog-Register sind analog mit den beschriebenen MODBUS-Funktionscodes (read und write) ansprechbar. Statt der Adresse eines Klemmenkanals wird dazu die jeweilige Registeradresse angegeben.

Tabelle 121: Registeradresse 0x1000

Registeradresse 0x1000 (4096_{dez})	
Wert	Watchdog time, WS_TIME
Zugang	Lesen/schreiben
Standard	0x0064
Beschreibung	Dieses Register speichert den Wert für die Zeitüberschreitung (Time-out). Damit der Watchdog gestartet werden kann, muss der Vorgabewert auf einen Wert ungleich Null geändert werden. Die Zeit wird in Vielfachen von 100 ms gesetzt, 0x0009 bedeutet also eine Time-out-Zeit von 0.9 s. Dieser Wert kann bei laufendem Watchdog nicht geändert werden. Es gibt keinen Code, durch den der aktuelle Datenwert nochmals geschrieben werden kann, während der Watchdog aktiv ist.

Tabelle 122: Registeradresse 0x1001

Registeradresse 0x1001 (4097_{dez})	
Wert	Watchdog-Funktion Codiermaske, Funktionscode 1...16, WDFCM_1_16
Zugang	Lesen/schreiben
Standard	0xFFFF
Beschreibung	Mittels dieser Maske sind die Funktionscodes einstellbar, um die Watchdog-Funktion zu triggern. Mit einer „1“ an den folgend beschriebenen Bitpositionen kann der Funktionscode ausgewählt werden: FC 1 Bit 0 FC 2 Bit 1 FC 3 Bit 2 FC 4 Bit 3 FC 5 Bit 4 ... FC 16 Bit 15 Der Registerwert kann nur geändert werden, wenn der Watchdog deaktiviert ist. Das im Register gespeicherte Bitmuster gibt an, welche Funktionscodes zum Auslösen des Watchdogs führen. Einige Funktionscodes werden nicht unterstützt. Für diese können zwar Werte eingetragen werden, diese starten den Watchdog jedoch nicht, auch nicht, wenn ein anderes MODBUS-Gerät diese sendet.

Tabelle 123: Registeradresse 0x1002

Registeradresse 0x1002 (4098_{dez})	
Wert	Watchdog-Funktion Codiermaske, Funktionscode 17...32, WD_FCM_17_32
Zugang	Lesen/schreiben
Standard	0xFFFF
Beschreibung	Gleiche Funktion wie zuvor, aber mit den Funktionscodes 17 bis 32. FC 17 Bit 0 FC 18 Bit 1 ... FC 32 Bit 15 Diese Codes werden nicht unterstützt. Dieses Register sollte deshalb auf dem Vorgabewert belassen werden. Der Registerwert kann nur geändert werden, wenn der Watchdog deaktiviert ist. Es gibt keinen Ausnahmecode durch den der aktuelle Datenwert nochmals geschrieben werden kann, während der Watchdog aktiv ist.

Tabelle 124: Registeradresse 0x1003

Registeradresse 0x1003 (4099_{dez})	
Wert	Watchdog-Trigger, WD_TRIGGER
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Dieses Register wird für eine alternative Trigger-Methode benutzt. Durch das Schreiben unterschiedlicher Werte in dieses Register wird der Watchdog getriggert. Aufeinanderfolgende Werte müssen sich in der Größe unterscheiden. Das Schreiben eines Werts ungleich Null startet den Watchdog nach einem Power-On. Für einen Neustart muss der geschriebene Wert unbedingt ungleich dem vorher geschriebenen sein! Ein Watchdog-Fehler wird zurückgesetzt und das Schreiben der Prozessdaten wird wieder ermöglicht.

Tabelle 125: Registeradresse 0x1004

Registeradresse 0x1004 (4100_{dez})	
Wert	Minimale aktuelle Trigger-Zeit, WD_AC_TRG_TIME
Zugang	Lesen
Standard	0xFFFF
Beschreibung	Dieses Register speichert die aktuell kleinste Watchdog-Trigger-Zeit. Bei einem Triggern des Watchdogs, wird der gespeicherte Wert mit dem aktuellen verglichen. Ist der aktuelle Wert kleiner als der gespeicherte, wird dieser durch den aktuellen Wert ersetzt. Die Einheit ist 100 ms/Digit. Durch das Schreiben neuer Werte wird der gespeicherte Wert geändert. Dies hat keine Auswirkung auf den Watchdog. Der Wert 0x0000 ist nicht erlaubt.

Tabelle 126: Registeradresse 0x1005

Registeradresse 0x1005 (4101_{dez})	
Wert	Watchdog stoppen, WD_AC_STOP_MASK
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Wird der Wert 0xAAAA gefolgt von dem Wert 0x5555 in dieses Register geschrieben, stoppt der Watchdog. Die Watchdog-Fehlerreaktion wird gesperrt. Ein Watchdog-Fehler wird zurückgesetzt und das Schreiben auf die Prozessdaten wird wieder ermöglicht.

Tabelle 127: Registeradresse 0x1006

Registeradresse 0x1006 (4102_{dez})	
Wert	Während Watchdog läuft, WD_RUNNING
Zugang	Lesen
Standard	0x0000
Beschreibung	Aktueller Watchdog-Status bei 0x0000: Watchdog nicht aktiv bei 0x0001: Watchdog aktiv bei 0x0002: Watchdog abgelaufen

Tabelle 128: Registeradresse 0x1007

Registeradresse 0x1007 (4103_{dez})	
Wert	Watchdog neu starten, WD_RESTART
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Schreiben von 0x1 in das Register startet den Watchdog wieder. Wurde der Watchdog vor dem Überlauf gestoppt, wird er nicht wieder gestartet.

Tabelle 129: Registeradresse 0x1008

Registeradresse 0x1008 (4104_{dez})	
Wert	Watchdog einfach anhalten, WD_AC_STOP_SIMPLE
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Durch Schreiben der Werte 0xAA55 oder 0x55AA wird der Watchdog angehalten, falls er aktiv war. Die Watchdog-Fehlerreaktion wird vorübergehend deaktiviert. Ein anstehender Watchdog-Fehler wird zurückgesetzt und ein Schreiben ins Watchdog-Register ist wieder möglich.

Tabelle 130: Registeradresse 0x1009

Registeradresse 0x1009 (4105_{dez})	
Wert	MODBUS-Socket nach Watchdog-Time-out schließen
Zugang	Lesen/schreiben
Beschreibung	0: MODBUS-Socket wird nicht geschlossen 1: MODBUS-Socket wird geschlossen

Tabelle 131: Registeradresse 0x100A

Registeradresse 0x100A (4106_{dez})	
Wert	Alternativer Watchdog
Zugang	Lesen/schreiben
Standard	0x0000
Beschreibung	Schreiben eines Zeitwertes in Register 0x1000 Register 0x100A = 0x0001: Watchdog wird aktiv geschaltet Mit dem ersten MODBUS-Telegramm wird der Watchdog gestartet. Der Watchdog wird mit jedem MODBUS/TCP-Befehl getriggert. Nach Ablauf der Watchdog-Zeit werden alle Ausgänge auf Null gesetzt. Die Ausgänge können durch erneutes Schreiben wieder gesetzt werden. Das Register 0x100A ist remanent und damit auch das Register 0x1000. Bei eingeschaltetem Watchdog lässt sich der Zeitwert in Register 0x1000 nicht mehr ändern.

Die Länge jedes Registers beträgt 1 Wort, d. h. bei jedem Zugriff kann lediglich ein Wort geschrieben oder gelesen werden. Im Folgenden werden zwei Beispiele zum Setzen des Wertes für die Zeitüberschreitung aufgeführt:

Watchdog für eine Zeitüberschreitung von 1 Sekunde oder mehr setzen:

1. Schreiben Sie 0x000A in das Register für Zeitüberschreitung (0x1000). (Register 0x1000 arbeitet mit Vielfachen von 100 ms; 1 s = 1000 ms; 1000 ms / 100 ms = $10_{dez} = A_{hex}$)

2. Starten Sie mittels des Funktionscodes 5 den Watchdog, indem Sie 0x0010 ($=2^{(5-1)}$) in die Codiermaske (Register 0x1001) schreiben.

Tabelle 132: Watchdog starten

FC	FC16	FC15	FC14	FC13	FC12	FC11	FC10	FC9	FC8	FC7	FC6	**FC5**	FC4	FC3	FC2	FC1
Bit	15	14	13	12	11	10	9	8	7	6	5	**4**	3	2	1	0
bin	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0
hex	0				0				**1**				0			

Der Funktionscode 5 (Schreiben eines digitalen Ausgangsbits) triggert den Watchdog kontinuierlich, um den Watchdog-Timer innerhalb der angegebenen Zeit immer wieder neu zu starten. Wird zwischen den Anfragen mehr als 1 Sekunde erreicht, ist ein Watchdog-Time-out-Fehler aufgetreten.

3. Um den Watchdog zu stoppen, schreiben Sie den Wert 0xAA55 oder 0x55AA in das Register 0x1008 (Watchdog einfach anhalten, WD_AC_STOP_SIMPLE).

Watchdog für eine Zeitüberschreitung von 10 Minuten oder mehr setzen

1. Schreiben Sie 0x1770 (= 10*60*1000 ms / 100 ms) in das Register für Zeitüberschreitung (0x1000).
   (Register 0x1000 arbeitet mit Vielfachen von 100 ms;
   10 min = 600.000 ms; 600.000 ms / 100 ms = 6000_{dez} = 1770_{hex})

2. Starten Sie den Watchdog, indem Sie 0x0001 in den Watchdog-Trigger-Register (0x1003) schreiben.

3. Um den Watchdog zu triggern, schreiben Sie unterschiedliche Werte, z. B. Zählwerte 0x0000, 0x0001 etc. in das Watchdog-Trigger-Register (0x1003).

Die nacheinander geschriebenen Werte müssen sich in der Größe unterscheiden. Das Schreiben eines Wertes ungleich Null startet den Watchdog. Watchdog-Fehler werden zurückgesetzt und das Schreiben der Prozessdaten wird wieder ermöglicht.

4. Um den Watchdog zu stoppen, schreiben Sie den Wert 0xAA55 oder 0x55AA in das Register 0x1008 (Watchdog einfach anhalten, WD_AC_STOP_SIMPLE).

Tabelle 133: Registeradresse 0x100B

Registeradresse 0x100B (4107_{dez})	
Wert	Save-Watchdog-Parameter
Zugang	Schreiben
Standard	0x0000
Beschreibung	Mit Schreiben von „0x55AA“ oder „0xAA55“ in Register 0x100B werden die Register 0x1000, 0x1001, 0x1002 auf „remanent“ gesetzt.

12.2.5.3 Diagnoseregister

Folgende Register können gelesen werden, um einen Fehler des Feldbusknotens zu bestimmen:

Tabelle 134: Registeradresse 0x1020

Registeradresse 0x1020 (4128$_{dez}$)	
Wert	LedErrCode
Zugang	Lesen
Beschreibung	Angabe des Fehlercodes

Tabelle 135: Registeradresse 0x1021

Registeradresse 0x1021 (4129$_{dez}$)	
Wert	LedErrArg
Zugang	Lesen
Beschreibung	Angabe des Fehlerargumentes

12.2.5.4 Konfigurationsregister

Folgende Register können gelesen werden, um die Konfiguration der angeschlossenen Busklemmen zu bestimmen:

Tabelle 136: Registeradresse 0x1022

Registeradresse 0x1022 (4130_{dez})	
Wert	CnfLen.AnalogOut
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenworten der Ausgänge

Tabelle 137: Registeradresse 0x1023

Registeradresse 0x1023 (4131_{dez})	
Wert	CnfLen.AnalogInp
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenworten der Eingänge

Tabelle 138: Registeradresse 0x1024

Registeradresse 0x1024 (4132_{dez})	
Wert	CnfLen.DigitalOut
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenbits der Ausgänge

Tabelle 139: Registeradresse 0x1025

Registeradresse 0x1025 (4133_{dez})	
Wert	CnfLen.DigitalInp
Zugang	Lesen
Beschreibung	Anzahl E/A-Bits bei den Prozessdatenbits der Eingänge

Tabelle 140: Registeradresse 0x1028

Registeradresse 0x1028 (4136_{dez})	
Wert	Bootoptions
Zugang	Lesen/schreiben
Beschreibung	Bootkonfiguration: 1: BootP 2: DHCP 3: BootP-Request before static IP 4: EEPROM

Tabelle 141: Registeradresse 0x1029

Registeradresse 0x1029 (4137_{dez}) mit bis zu 9 Worten	
Wert	MODBUS/TCP-Statistik
Zugang	Lesen/schreiben
Beschreibung	1 Wort SlaveDeviceFailure → Klemmenbusfehler, Feldbusfehler bei eingeschaltetem Watchdog 1 Wort BadProtocol → Fehler im MODBUS/TCP-Header 1 Wort BadLength → Falsche Telegrammlänge 1 Wort BadFunction → Ungültiger Funktionscode 1 Wort Bad Address → Ungültige Registeradresse 1 Wort BadData → Ungültiger Wert 1 Wort TooManyRegisters → Anzahl der zu bearbeitenden Register zu groß, Lesen/Schreiben 125/100 1 Wort TooManyBits → Anzahl der zu bearbeitenden Coils zu groß, Lesen/Schreiben 2000/800 1 Wort ModTcpMessageCounter → Anzahl der empfangenen MODBUS/TCP-Telegramme Durch Schreiben von 0xAA55 oder 0x55AA wird das Register zurückgesetzt.

Tabelle 142: Registeradresse 0x102A

Registeradresse 0x102A (4138_{dez}) mit bis zu 1 Wort	
Wert	MODBUS/TCP-Connections
Zugang	Lesen
Beschreibung	Anzahl der TCP-Verbindungen

Tabelle 143: Registeradresse 0x102B

Registeradresse 0x102B (4139_{dez}) mit bis zu 1 Wort	
Wert	KBUS-Reset
Zugang	Schreiben
Beschreibung	Schreiben auf dieses Register löst einen Reset des Klemmenbus aus.

Tabelle 144: Registeradresse 0x1030

Registeradresse 0x1030 (4144_{dez}) mit bis zu 1 Wort	
Wert	Konfiguration MODBUS/TCP-Time-out
Zugang	Lesen/schreiben
Standard	0x0258 (600 dezimal)
Beschreibung	Dieses Register speichert den Wert für eine TCP-Verbindungsüberwachung. Der Standardwert ist 600 ms (60 Sekunden), die Zeitbasis ist 100 ms, der Minimalwert ist 100 ms. Geöffnete TCP-Verbindungen werden automatisch geschlossen, wenn die eingetragene Zeit je Verbindung überschritten wurde. Wird der Wert auf ‚0' gesetzt, ist der Watchdog nicht aktiv. Der Watchdog wird mit einer Anfrage auf der Verbindung getriggert.

Tabelle 145: Registeradresse 0x1031

Registeradresse 0x1031 (4145_{dez}) mit bis zu 3 Worten	
Wert	Lesen der MAC-ID des Feldbuskopplers/-controllers
Zugang	Lesen
Beschreibung	Ausgabe der MAC-ID, Länge 3 Worte

Tabelle 146: Registeradresse 0x1035

Registeradresse 0x1035 (4149_{dez}) 1 Wort	
Wert	Konfiguration des Timeoffsets zur GMT Zeit
Zugang	Lesen / schreiben
Standard	0x0000
Beschreibung	In diesem Register kann der Zeit-Offset zur GMT-Zeit eingestellt werden. Die Offset-Einstellung ist in dem Bereich von -12 bis +12 Stunden möglich.

Tabelle 147: Registeradresse 0x1036

Registeradresse 0x1036 (4150_{dez}) 1 Wort	
Wert	Konfiguration der Sommer- oder Winterzeit
Zugang	Lesen / schreiben
Standard	0x0000
Beschreibung	In diesem Register kann die Sommer- oder Winterzeit (Daylight Saving Time) eingestellt werden. Für die Einstellung ist der Wert 0 und 1 zulässig.

Tabelle 148: Registeradresse 0x1037

Registeradresse 0x1031 (4151_{dez}) mit bis zu 3 Worten	
Wert	Konfiguration der Modbus Response Delay Zeit
Zugang	Lesen / schreiben
Standard	0x0000
Beschreibung	Dieses Register speichert den Wert der Modbus Response Delay Zeit für eine Modbus TCP Verbindung. Die Zeitbasis ist 1 ms. Bei einer bestehenden Modbus TCP Verbindung wird die Response um die eingetragene Zeit verzögert.

Tabelle 149: Registeradresse 0x1050

Registeradresse 0x1050 (4176_{dez})	**ab Firmwarestand 9**
Wert	Diagnose angeschlossener Busklemmen
Zugang	Lesen
Beschreibung	Diagnose angeschlossener Busklemmen, Länge 3 Worte Wort 1: Busklemmennummer Wort 2: Kanalnummer Wort 3: Diagnose

Tabelle 150: Registeradresse 0x2030

Registeradresse 0x2030 (8240_{dez}) mit bis zu 65 Worten																
Wert	Beschreibung der angeschlossenen Busklemmen															
Zugang	Lesen der Busklemmen 0...64															
Beschreibung	Länge 1...65 Worte Über Register 0x2030 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Bestellnummer der Busklemmen bzw. des Feldbuskopplers/ -controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Bestellnummern von digitalen Busklemmen nicht ausgelesen werden können, wird eine digitale Busklemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Busklemmengröße in Bit Bitposition 15 → Kennung digitale Busklemme															
	Beispiele:															
	4-Kanal-Digitaleingangsklemme = 0x8401															
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Code	1	0	0	0	0	1	0	0	0	0	0	0	0	0	0	1
	Hex	8				4				0				1			
	2-Kanal-Digitalausgangsklemme = 0x8202																
	Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Code	1	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
	Hex	8				2				0				2			

Tabelle 151: Registeradresse 0x2031

Registeradresse 0x2031 (8241_{dez}) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Busklemmen
Zugang	Lesen der Busklemmen 65...128
Beschreibung	Länge 1...64 Worte Über Register 0x2031 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Bestellnummer der Busklemmen bzw. des Feldbuskopplers/ -controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Bestellnummern von digitalen Busklemmen nicht ausgelesen werden können, wird eine digitale Busklemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung : Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Busklemmengröße in Bit Bitposition 15 → Kennung digitale Busklemme

Tabelle 152: Registeradresse 0x2032

Registeradresse 0x2032 (8242_{dez}) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Busklemmen
Zugang	Lesen der Busklemmen 129...192
Beschreibung	Länge 1...64 Worte Über Register 0x2032 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Bestellnummer der Busklemmen bzw. des Feldbuskopplers/ -controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Bestellnummern von digitalen Busklemmen nicht ausgelesen werden können, wird eine digitale Busklemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Busklemmengröße in Bit Bitposition 15 → Kennung digitale Busklemme

Tabelle 153: Registeradresse 0x2033

Registeradresse 0x2033 (8243_{dez}) mit bis zu 65 Worten	
Wert	Beschreibung der angeschlossenen Busklemmen
Zugang	Lesen der Busklemmen 193...255
Beschreibung	Länge 1...63 Worte Über Register 0x2033 kann die Konfiguration des Knotens ermittelt werden. Dabei wird die Bestellnummer der Busklemmen bzw. des Feldbuskopplers/ -controllers (ohne führende 750) der Reihe nach aufgelistet. Jede Bezeichnung wird in einem Wort dargestellt. Da Bestellnummern von digitalen Busklemmen nicht ausgelesen werden können, wird eine digitale Busklemme codiert dargestellt. Die einzelnen Bits haben dann die folgende Bedeutung: Bitposition 0 → Eingangsklemme Bitposition 1 → Ausgangsklemme Bitposition 2...7 → nicht benutzt Bitposition 8...14 → Busklemmengröße in Bit Bitposition 15 → Kennung digitale Busklemme

Tabelle 154: Registeradresse 0x2040

Registeradresse 0x2040 (8256_{dez})	
Wert	Ausführen eines Software-Resets
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Durch Schreiben der Werte 0xAA55 oder 0x55AA führt der Feldbuskoppler/-controller einen Neustart durch.

Tabelle 155: Registeradresse 0x2041

Registeradresse 0x2041 (8257_{dez})	
Wert	Flash-Format
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Das Flash-Dateisystem wird neu formatiert

Tabelle 156: Registeradresse 0x2042

Registeradresse 0x2042 (8258_{dez})	
Wert	Dateien extrahieren
Zugang	Schreiben (Schreibsequenz 0xAA55 oder 0x55AA)
Beschreibung	Die Standarddateien (HTML-Seiten) des Feldbuskopplers/-controllers werden extrahiert und in das Flash geschrieben.

Tabelle 157: Registeradresse 0x2043

Registeradresse 0x2043 (8259_{dez})	
Wert	0x55AA
Zugang	Schreiben
Beschreibung	Werkseinstellungen Die Werkseinstellungen sind nach dem nächsten Reset des Gerätes wirksam, z. B. SW-Reset über MODBUS-Registeradresse 0x2040.

12.2.5.5 Firmware-Informationsregister

Folgende Register werden genutzt, um Informationen zur Firmware des Feldbuskopplers/-controllers auszulesen:

Tabelle 158: Registeradresse 0x2010

Registeradresse 0x2010 (8208_{dez}) mit bis zu 1 Wort	
Wert	Revision, INFO_REVISION
Zugang	Lesen
Beschreibung	Firmware-Index, z. B. 0x0005 für Version 5

Tabelle 159: Registeradresse 0x2011

Registeradresse 0x2011 (8209_{dez}) mit bis zu 1 Wort	
Wert	Series code, INFO_SERIES
Zugang	Lesen
Beschreibung	WAGO-Baureihennummer, z. B. 0x02EE (750 dez.) für WAGO-I/O-SYSTEM 750

Tabelle 160: Registeradresse 0x2012

Registeradresse 0x2012 (8210_{dez}) mit bis zu 1 Wort	
Wert	Item number, INFO_ITEM
Zugang	Lesen
Beschreibung	WAGO-Bestellnummer, z. B. 0x0349 (841 dez.) für den Controller 750-841, 0x0155 (341 dec.) für den Feldbuskoppler 750-341 etc.

Tabelle 161: Registeradresse 0x2013

Registeradresse 0x2013 (8211_{dez}) mit bis zu 1 Wort	
Wert	Major sub item code, INFO_MAJOR
Zugang	Lesen
Beschreibung	Firmware-Version Major-Revision

Tabelle 162: Registeradresse 0x2014

Registeradresse 0x2014 (8212_{dez}) mit bis zu 1 Wort	
Wert	Minor sub item code, INFO_MINOR
Zugang	Lesen
Beschreibung	Firmware-Version Minor-Revision

Tabelle 163: Registeradresse 0x2020

Registeradresse 0x2020 (8224_{dez}) mit bis zu 16 Worten	
Wert	Description, INFO_DESCRIPTION
Zugang	Lesen
Beschreibung	Informationen zum Feldbuskoppler/-controller, 16 Worte

Tabelle 164: Registeradresse 0x2021

Registeradresse 0x2021 (8225_{dez}) mit bis zu 8 Worten	
Wert	Description, INFO_DESCRIPTION
Zugang	Lesen
Beschreibung	Zeit des Firmwarestandes, 8 Worte

Tabelle 165: Registeradresse 0x2022

Registeradresse 0x2022 (8226_{dez}) mit bis zu 8 Worten	
Wert	Description, INFO_DATE
Zugang	Lesen
Beschreibung	Datum des Firmwarestandes, 8 Worte

Tabelle 166: Registeradresse 0x2023

Registeradresse 0x2023 (8227_{dez}) mit bis zu 32 Worten	
Wert	Description, INFO_LOADER_INFO
Zugang	Lesen
Beschreibung	Info über Programmierung der Firmware, 32 Worte

12.2.5.6 Konstantenregister

Folgende Register enthalten Konstanten, die genutzt werden können, um die Kommunikation mit dem Master zu testen:

Tabelle 167: Registeradresse 0x2000

Registeradresse 0x2000 (8192_{dez})	
Wert	Null, GP_ZERO
Zugang	Lesen
Beschreibung	Konstante mit Null

Tabelle 168: Registeradresse 0x2001

Registeradresse 0x2001 (8193_{dez})	
Wert	Einsen, GP_ONES
Zugang	Lesen
Beschreibung	Konstante mit Einsen. • „-1“, wenn Konstante als „signed int“ deklariert ist • „MAXVALUE“, wenn Konstante als „unsigned int“ deklariert ist

Tabelle 169: Registeradresse 0x2002

Registeradresse 0x2002 (8194_{dez})	
Wert	1,2,3,4, GP_1234
Zugang	Lesen
Beschreibung	Konstanter Wert, zum Testen, ob High- und Low-Byte getauscht sind (Intel/Motorola Format). Sollte im Master als 0x1234 erscheinen. Erscheint 0x3412, müssen High- und Low-Byte getauscht werden.

Tabelle 170: Registeradresse 0x2003

Registeradresse 0x2003 (8195_{dez})	
Wert	Maske 1, GP_AAAA
Zugang	Lesen
Beschreibung	Konstante, die anzeigt, ob alle Bits vorhanden sind. Wird zusammen mit Register 0x2004 genutzt.

Tabelle 171: Registeradresse 0x2004

Registeradresse 0x2004 (8196_{dez})	
Wert	Maske 1, GP_5555
Zugang	Lesen
Beschreibung	Konstante, die anzeigt, ob alle Bits vorhanden sind. Wird zusammen mit Register 0x2003 genutzt.

Tabelle 172: Registeradresse 0x2005

Registeradresse 0x2005 (8197_{dez})	
Wert	Größte positive Zahl, GP_MAX_POS
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 173: Registeradresse 0x2006

Registeradresse 0x2006 (8198_{dez})	
Wert	Größte negative Zahl, GP_MAX_NEG
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 174: Registeradresse 0x2007

Registeradresse 0x2007 (8199_{dez})	
Wert	Größte halbe positive Zahl, GP_HALF_POS
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 175: Registeradresse 0x2008

Registeradresse 0x2008 (8200_{dez})	
Wert	Größte halbe negative Zahl, GP_HALF_NEG
Zugang	Lesen
Beschreibung	Konstante, um die Arithmetik zu kontrollieren

Tabelle 176: Registeradresse 0x3000 bis 0x4FFF

Registeradresse 0x3000 bis 0x4FFF (12288_{dez} bis 20479_{dez})	
Wert	Retain-Bereich
Zugang	Lesen/schreiben
Beschreibung	In diesen Registern kann auf den Merker/Retain-Bereich zugegriffen werden.

12.3 EtherNet/IP (Ethernet/Industrial Protocol)

12.3.1 Allgemeines

EtherNet/IP steht für „Ethernet Industrial Protocol“ und definiert einen offenen Industrie-Standard, der das klassische Ethernet mit einem Industrie-Protokoll erweitert. Dieser Standard wurde gemeinsam von der „ControlNet International“ (CI) und der „Open DeviceNet Vendor Association“ (ODVA) entwickelt mit Unterstützung der „Industrial Ethernet Association“ (IEA).

Durch dieses Kommunikationssystem wird es Geräten ermöglicht, zeitkritische Applikationsdaten in einer industriellen Umgebung auszutauschen. Das Gerätespektrum reicht von einfachen I/O-Geräten (z. B. Sensoren) bis zu komplexen Steuerungen (z. B. Roboter).

EtherNet/IP basiert auf der TCP/IP-Protokoll-Familie und übernimmt somit die unteren 4 Schichten des OSI-7-Schichten-Modells in unveränderter Form, so dass alle Standard-ETHERNET-Kommunikationsmodule, wie z. B. Interface-Karten für PC, Kabel, Konnektoren, Hubs und Switches mit EtherNet/IP gleichfalls verwendet werden können.
Oberhalb der Transport-Schicht befindet sich das „Encapsulation Protocol“, mit dem das „Common Industrial Protocol“ (CIP) auf TCP/IP und UDP/IP aufgesetzt ist.

CIP, als ein großer netzwerkunabhängiger Standard, wird bereits bei ControlNet und DeviceNet benutzt. Die Überführung einer Applikation auf eines dieser Systeme ist somit sehr einfach realisierbar. Der Datenaustausch basiert auf einem Objektmodell.

ControlNet, DeviceNet und EtherNet/IP haben auf diese Weise dasselbe Applikationsprotokoll und können deshalb gemeinsam Geräteprofile und Objekt-Libraries nutzen. Diese Objekte machen eine Plug-and-play-Interoperabilität zwischen komplexen Geräten verschiedener Hersteller möglich.

12.3.2 Protokollübersicht im OSI-Modell

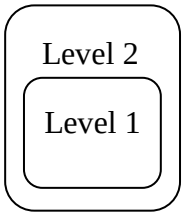
Zur Verdeutlichung der Gemeinsamkeiten zwischen DeviceNet, ControlNet und EtherNet/IP zeigt die folgende Darstellung die Einordnung der Protokolle in das 7-schichtige OSI-Referenzmodell (Open Systems Interconnection Reference Model).

Tabelle 177: OSI-Referenzmodell

<table>
<tr><td>7
Application
Layer</td><td colspan="3">Object Libary
(Communications, Applications, Time Synchronization)</td><td>Safety Object Libary</td><td rowspan="3">Comon Industrial Protocol (CIP)</td></tr>
<tr><td>6
Presentation
Layer</td><td colspan="3">Data Management Services
Explicit and I/O Messages</td><td>Safety Services and Messages</td></tr>
<tr><td>5
Session
Layer</td><td colspan="4">Connection Management, Routing</td></tr>
<tr><td>4
Transport
Layer</td><td>TCP/UDP</td><td rowspan="2">CompoNet
Network and Transport</td><td rowspan="2">ControlNet
Network
and
Transport</td><td rowspan="2">DeviceNet
Network and
Transport</td><td rowspan="4">Network Adaptations of CIP</td></tr>
<tr><td>3
Network
Layer</td><td>Internet
Protocol</td></tr>
<tr><td>2
Data Link
Layer</td><td>ETHERNET
CSMA/CD</td><td>CompoNet
Time Slot</td><td>ControlNet
CTDMA</td><td>CAN
CSMA/NBA</td></tr>
<tr><td>1
Physical
Layer</td><td>ETHERNET</td><td>CompoNet</td><td>ControlNet</td><td>DeviceNet</td></tr>
</table>

12.3.3 Eigenschaften der EtherNet/IP-Protokollsoftware

Die EtherNet/IP-Produktklassen sind in insgesamt 4 Level aufgeteilt, wobei jeder eine gewisse Funktionalität beinhaltet. Jeder höhere Level wiederum beinhaltet mindestens eine Funktionalität eines niedrigeren Levels. Der Feldbuskoppler/ -controller unterstützt die Level 1 und 2 der EtherNet/IP-Produktklassen, die unmittelbar aufeinander aufbauen.



Level 2: Level 1 + I/O Messages – Server

Level 1: Explicit Messages – Server

- UCMM fähig (verbindungslos, Client und Server)
- 128 „Encapsulation Protocol Sessions“
- 128 Klasse 3-Verbindungen oder Klasse 1 (kombiniert)

  Klasse 3-Verbindung – explizite Nachrichten
  (verbindungsorientiert, Client und Server)

  Klasse 1-Verbindung – I/O Nachrichten
  (verbindungsorientiert, Client und Server)

12.3.4 EDS-Datei

Die „Electronic Data Sheets“-Datei, kurz: EDS-Datei, enthält die Kenndaten des Feldbuskopplers/-controllers und Angaben zu seinen Kommunikationsfähigkeiten. Die für den EtherNet/IP-Betrieb benötigte EDS-Datei wird von der jeweiligen Projektierungssoftware eingelesen bzw. installiert.

Hinweis

Download der EDS-Datei!
Sie erhalten die EDS-Datei im Download-Bereich der WAGO-Internetseiten unter: www.wago.com.

Information

Information zur Installation der EDS-Datei
Entnehmen Sie bitte Hinweise zur Installation der EDS-Datei der Dokumentation zu der von Ihnen genutzten Projektierungssoftware.

12.3.5 Objektmodell

12.3.5.1 Allgemeines

Für die Netzwerkkommunikation verwendet EtherNet/IP ein Objektmodell, in dem alle Funktionen und Daten eines Gerätes beschrieben sind.
Jeder Knoten im Netz wird als Sammlung von Objekten dargestellt.
Das Objektmodell enthält Begriffe, die folgendermaßen definiert sind:

Objekt (object):
Ein Objekt ist eine abstrakte Darstellung von einzelnen, zusammengehörigen Bestandteilen innerhalb eines Gerätes. Es ist bestimmt durch seine Daten oder Eigenschaften (Attributes), seine nach außen bereitgestellten Funktionen oder Dienste (Services) und durch sein definiertes Verhalten (Behaviour).

Klasse (class):
Eine Klasse beschreibt eine Reihe von Objekten, die alle die gleiche Art von Systemkomponenten darstellen. Eine Klasse ist die Verallgemeinerung eines Objektes. Alle Objekte in einer Klasse sind in Bezug auf ihre Form und ihr Verhalten identisch, wobei sie jedoch unterschiedliche Attributwerte umfassen können.

Instanz (instance):
Eine Instanz beschreibt eine spezifische und tatsächliche (physikalische) Ausprägung eines Objektes. Die Benennungen „Objekt", „Instanz" und „Objektinstanz" beziehen sich alle auf eine spezifische Instanz.
Unterschiedliche Instanzen einer Klasse haben die gleichen Dienste (services), das gleiche Verhalten (behaviour) und die gleichen Variablen (attributes). Sie können jedoch unterschiedliche Variablenwerte haben.
Beispiel: Eine Instanz der Objektklasse „Land" ist beispielsweise Finnland.

Variable (attribute):
Die Variablen (attributes) beschreiben ein externes sichtbares Merkmal oder die Funktion eines Objektes. Typische Attribute sind beispielsweise Konfigurations- oder Statusinformationen.
Beispiel: Es wird der ASCII-Name eines Objektes oder die Wiederholungsfrequenz eines periodischen Objektes ausgegeben.

Dienst (service):
Ein Dienst ist eine Funktion, die von einem Objekt und/oder einer Objekt-Klasse unterstützt wird. CIP definiert eine Gruppe gemeinsamer Dienste, die auf die Variablen (Attribute) angewendet werden. Diese Dienste führen festgelegte Aktionen durch.
Beispiel: Das Lesen von Variablen.

Verhalten (behaviour):
Das Verhalten legt fest, wie ein Objekt funktioniert. Die Funktionen resultieren aus unterschiedlichen Ereignissen, die das Objekt ermittelt, wie zum Beispiel der Empfang von Serviceanforderungen, die Erfassung interner Störungen oder der Ablauf von Zeitnehmern.

12.3.5.2 Klassen-Übersicht

Die CIP-Klassen sind in der CIP-Spezifikation der ODVA enthalten. Sie beschreiben, unabhängig von der physikalischen Schnittstelle, z. B. ETHERNET, CAN, deren Eigenschaften (Band 1 „Common Industrial Protocol“). Die physikalische Schnittstelle wird in einer weiteren Spezifikation beschrieben. Für EtherNet/IP ist das der Band 2 („EtherNet/IP Adaption of CIP“), welcher die Anpassung des EtherNet/IP an CIP beschreibt.

WAGO nutzt hierbei die Klassen 01_{hex}, 02_{hex}, 04_{hex}, 05_{hex}, 06_{hex} und $F4_{hex}$, welche in Band 1 („Common Industrial Protocol“) beschrieben sind.
Aus dem Band 2 („EtherNet/IP Adaption of CIP“) werden die Klassen $F5_{hex}$ und $F6_{hex}$ unterstützt.

Darüber hinaus stehen WAGO-spezifische Klassen zur Verfügung, die in der unten stehenden Übersichtstabelle aufgeführt sind.

Alle gelisteten CIP-Common-Klassen und im Anschluss daran die WAGO-spezifischen Klassen werden, nach einer kurzen Erläuterung der Tabellenköpfe in den Objektbeschreibungen, in den folgenden einzelnen Kapiteln näher beschrieben.

Tabelle 178: Übersicht CIP-Common-Klassen

Klasse	Name
01 hex	Identity
02 hex	Message Router
04 hex	Assembly
05 hex	Connection
06 hex	Connection Manager
F4 hex	Port Class Object
F5 hex	TCP/IP Interface Object
F6 hex	Ethernet Link Object

Tabelle 179: Übersicht WAGO-spezifische Klassen

Klasse	Name
64 hex	Coupler/Controller Configuration Object
65 hex	Discrete Input Point
66 hex	Discrete Output Point
67 hex	Analog Input Point
68 hex	Analog Output Point
69 hex	Discrete Input Point Extended 1
6A hex	Discrete Output Point Extended 1
6B hex	Analog Input Point Extended 1
6C hex	Analog Output Point Extended 1
6D hex	Discrete Input Point Extended 2
6E hex	Discrete Output Point Extended 2
6F hex	Analog Input Point Extended 2

70 hex	Analog Output Point Extended 2
71 hex	Discrete Input Point Extended 3
72 hex	Discrete Output Point Extended 3
73 hex	Analog Input Point Extended 3
74 hex	Analog Output Point Extended 3
80 hex	Module Configuration
81 hex	Module Configuration Extended 1
A0 hex	Input fieldbus variable USINT
A1 hex	Input fieldbus variable USINT Extended 1
A2 hex	Input fieldbus variable USINT Extended 2
A3 hex	Output fieldbus variable USINT
A4 hex	Output fieldbus variable USINT Extended 1
A5 hex	Output fieldbus variable USINT Extended 2
A6 hex	Input fieldbus variable UINT
A7 hex	Input fieldbus variable UINT Extended 1
A8 hex	Output fieldbus variable UINT
A9 hex	Output fieldbus variable UINT Extended 1
AA hex	Input fieldbus variable UDINT
AB hex	Input fieldbus variable UDINT Offset UINT
AC hex	Output fieldbus variable UDINT
AD hex	Output fieldbus variable UDINT Offset UINT

12.3.5.3 Erläuterung der Tabellenköpfe in den Objektbeschreibungen

Tabelle 180: Erläuterung der Tabellenköpfe in den Objektbeschreibungen

Spaltenüberschrift	Beschreibung
Attribut ID	Integerwert, der dem entsprechenden Attribut zugeordnet ist
Zugriff	**Set:** Auf das Attribut kann mittels des Dienstes Set_Attribute zugegriffen werden (Schreiben/Verändern des Attribut-Wertes).
	Hinweis **Ansprechen auch mit Get_Attribute-Dienst möglich!** Unterstützt ein Attribut den Dienst Set_Attribute, so kann dieses auch mit dem Dienst Get_Attribute angesprochen werden.
	Get: Auf das Attribut kann mittels Get_Attribute- Services zugegriffen werden (Lesen des Attribut-Wertes). **Get_Attribute_All:** Liefert den Inhalt aller Attribute. **Set_Attribute_Single:** Modifiziert einen Attribut-Wert. **Reset:** Führt einen Neustart durch. 0: Neustart 1: Neustart und Wiederherstellen der Werkseinstellungen

NV	**NV (non volatile):** Das Attribut wird permanent im Feldbuskoppler/-controller gespeichert.
	V (volatile): Das Attribut wird nicht permanent im Feldbuskoppler/-controller gespeichert.
	Hinweis **Ohne Angabe wird das Attribut nicht gespeichert!** Ist diese Spalte nicht vorhanden, sind alle Attribute vom Typ V (volatile).
Name	Bezeichnung des Attributs.
Datentyp	Bezeichnung des CIP-Datentyps des Attributes
Beschreibung	Kurze Beschreibung zu dem Attribut.
Defaultwert	Werkseinstellung.

12.3.5.4 Identity (01 $_{hex}$)

Die Klasse „Identity“ dient dazu, allgemeine Informationen des Feldbuskopplers/ -controllers bereitzustellen, die diesen eindeutig identifizieren.

Instanz 0 (Klassenattribute)

Tabelle 181: Identity (01 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Maximale Instanz	1 (0x0001)
3	Get	Max ID Number of Class Attributes	UINT	Maximale Anzahl der Klassen-Attribute (nur mit dem Dienst Get_Attribute_All)	0 (0x0000)
4	Get	Max ID Number of Instance Attribute	UINT	Maximale Anzahl der Instanz-Attribute (nur mit dem Dienst Get_Attribute_All)	0 (0x0000)

Instanz 1

Tabelle 182: Identity (01 $_{hex}$) – Instanz 1

Attribut ID	Zu-griff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Vendor ID	UINT	Hersteller-identifikation	40 (0x0028)
2	Get	Device Type	UINT	Generelle Typ-bezeichnung des Produktes	12 (0x000C)
3	Get	Product Code	UINT	Bezeichnung des Feldbus-kopplers/ -controllers	750-852 (in hex)
4	Get	Revision	STRUCT of:	Revision des Identity-Objektes	Firmware abhängig
		Major Revision	UINT		
		Minor Revision	UINT		
5	Get	Status	WORD	Aktueller Status des Gerätes	Bit 0 Zuweisung zu einem Master Bit 1 = 0 reserviert Bit 2 (konfiguriert) = 0 Konfiguration ist unverändert = 1 Konfiguration weicht von Hersteller-parametern ab Bit 3 = 0 reserviert Bit 4-7 Extended Device Status =0010 mind. eine fehlerhafte I/O-Verbindung =0011 keine I/O-Verbindung hergestellt Bit 8-11 nicht genutzt Bit 12-15 =0 reserviert
6	Get	Serial Number	UDINT	Seriennummer	die letzten 4 Stellen der MAC-ID
7	Get	Product Name	SHORT_ STRING	Produktname	ETHERNET ECO Controller 750-852

Common Services

Tabelle 183: Identity (01 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
05 $_{hex}$	Nein	Ja	Reset	Führt einen Neustart durch. 0: Neustart 1: Neustart und Wiederherstellen der Werkseinstellungen
0E $_{hex}$	Nein	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.5 Message Router (02 hex)

Das „Message Router Object“ stellt Verbindungspunkte in Form von Klassen oder Instanzen bereit, welche einen Client zum Adressieren von Diensten (Lesen, Schreiben) nutzen kann. Diese Nachrichten können sowohl verbindungsorientiert (connected) als auch verbindungslos (unconnected) vom Client an den Feldbuskoppler/-controller gesendet werden.

Instanz 0 (Klassenattribute)

Tabelle 184: Message Router (02 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Number of Attributes	UINT	Anzahl der Attribute	0 (0x0000)
3	Get	Number of Services	UINT	Anzahl der Dienste	0 (0x0000)
4	Get	Max ID Number of Class Attributes	UINT	Maximale Anzahl der Klassen-Attribute	0 (0x0000)
5	Get	Max ID Number of Instance Attributes	UINT	Maximale Anzahl der Instanz-Attribute	0 (0x0000)

Hinweis

Nur Dienst Get_Attribute_All anwendbar!
Die Klassen-Attribute sind nur mit dem Dienst Get_Attribute_All erreichbar.

Instanz 1

Tabelle 185: Message Router (02 hex) – Instanz 1

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	ObjectList	STRUCT of:	-	
		Number	UINT	Anzahl implementierter Klassen	40 (0x0028)
		Classes	UINT	Implementierte Klassen	01 00 02 00 04 00 06 00 F4 00 F5 00 F6 00 64 00 65 0066 0067 00 68 00 69 00 6A 00 6B 00 6C 00 6D 00 6E 00 6F 00 70 00 71 00 72 00 73 00 74 00 80 00 81 00 A0 00 A1 00 A2 00 A6 00 A7 00 AA 00 AB 00 A3 00 A4 00 A5 00 A8 00 A9 00 AC 00 AD 00
2	Get	NumberAvailable	UINT	Maximale Anzahl von unterschiedlichen Verbindungen	128 (0x0080)

Common Services

Tabelle 186: Message Router (02 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Nein	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E $_{hex}$	Nein	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.6 Assembly (04 $_{hex}$)

Mit Hilfe der „Assembly“-Klasse lassen sich mehrere auch verschiedenartige Objekte zusammenfassen. Diese können z. B. Ein- und Ausgangsdaten, Status- und Steuerinformationen oder Diagnoseinformationen sein. WAGO nutzt hier die herstellerspezifischen Instanzen, um diese Objekte in verschiedenen Anordnungen für Sie bereitzustellen. Hierdurch steht Ihnen ein effizienter Weg zum Austausch von Prozessdaten zur Verfügung. Im Folgenden werden die einzelnen statischen Assembly Instanzen mit deren Inhalten und Anordnungen beschrieben.

Instanz 0 (Klassenattribute)

Tabelle 187: Assembly (04 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	2 (0x0002)
2	Get	Max Instance	UINT	Höchste Instanz	111 (0x006F)

Übersicht der Instanzen

Tabelle 188: Statische Assembly-Instanzen – Übersicht

Instanz	Beschreibung
Instanz 101 (65 $_{hex}$)	für analoge und digitale Ausgangsdaten sowie Feldbus-Eingangsvariablen
Instanz 102 (66 $_{hex}$)	für digitale Ausgangsdaten und Feldbus-Eingangsvariablen
Instanz 103 (67 $_{hex}$)	für analoge Ausgangsdaten und Feldbus-Eingangsvariablen
Instanz 104 (68 $_{hex}$)	für analoge und digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 105 (69 $_{hex}$)	für digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 106 (6A $_{hex}$)	für analoge Eingangsdaten, Status und Feldbus-Ausgangsvariablen
Instanz 107 (6B $_{hex}$)	für digitale und analoge Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 108 (6C $_{hex}$)	für digitale Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 109 (6D $_{hex}$)	für analoge Eingangsdaten und Feldbus-Ausgangsvariablen
Instanz 110 (6E $_{hex}$)	für Feldbus-Ausgangsvariablen
Instanz 111 (6F $_{hex}$)	für Feldbus-Eingangsvariablen

Instanz 101 (65 $_{hex}$)

Diese Assembly-Instanz enthält analoge und digitale Ausgangsdaten. Eventuell definierte Feldbus-Eingangsvariablen werden hinter diesen angehängt.

Tabelle 189: Statische Assembly-Instanzen – Instanz 101 (65_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind analoge und digitale Ausgangsdaten sowie eventuell Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UNIT	Anzahl der Bytes im Prozessabbild	-

Instanz 102 (66_{hex})

Diese Assembly-Instanz enthält nur digitale Ausgangsdaten und Feldbus-Eingangsvariablen.

Tabelle 190: Statische Assembly-Instanzen – Instanz 102 (66_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind digitale Ausgangsdaten und Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 103 (67_{hex})

Diese Assembly-Instanz enthält nur analoge Ausgangsdaten und Feldbus-Eingangsvariablen.

Tabelle 191: Statische Assembly-Instanzen – Instanz 103 (67_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind analoge Ausgangsdaten und Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 104 (68_{hex})

Diese Assembly-Instanz enthält analoge und digitale Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 192: Statische Assembly-Instanzen – Instanz 104 (68_{hex})

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge und digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 105 (69 $_{hex}$)

Diese Assembly-Instanz enthält nur digitale Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 193: Statische Assembly-Instanzen – Instanz 105 (69 $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind digitale Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 106 (6A $_{hex}$)

Diese Assembly-Instanz enthält nur analoge Eingangsdaten, Status (= Wert aus Klasse 100, Instanz 1, Attribut 5) und Feldbus-Ausgangsvariablen.

Tabelle 194: Statische Assembly-Instanzen – Instanz 106 (6A $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge Eingangsdaten, Status und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 107 (6B $_{hex}$)

Diese Assembly-Instanz enthält analoge und digitale Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 195: Statische Assembly-Instanzen – Instanz 107 (6B $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge und digitale Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 108 (6C $_{hex}$)

Diese Assembly-Instanz enthält nur digitale Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 196: Statische Assembly-Instanzen – Instanz 108 (6C $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind digitale Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 109 (6D $_{hex}$)

Diese Assembly-Instanz enthält nur analoge Eingangsdaten und Feldbus-Ausgangsvariablen.

Tabelle 197: Statische Assembly-Instanzen – Instanz 109 (6D $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind analoge Eingangsdaten und Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 110 (6E $_{hex}$)

Diese Assembly-Instanz enthält Feldbus-Ausgangsvariablen.

Tabelle 198: Statische Assembly-Instanzen – Instanz 110 (6E $_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get	Data	ARRAY of BYTE	Es sind Feldbus-Ausgangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 111 (6F $_{hex}$)

Diese Assembly-Instanz enthält Feldbus-Eingangsvariablen.

Tabelle 199: Statische Assembly-Instanzen – Instanz 111 ($6F_{hex}$)

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
3	Get/Set	Data	ARRAY of BYTE	Es sind Feldbus-Eingangsvariablen im Prozessabbild enthalten	-
4	Get	Data Size	UINT	Anzahl der Bytes im Prozessabbild	-

Instanz 198 ($C6_{hex}$) „Input Only“

Mit dieser Instanz wird eine Verbindung aufgebaut, wenn keine Ausgänge angesprochen werden sollen bzw. wenn Eingänge abgefragt werden, die schon in einer Exclusive-Owner-Verbindung benutzt werden. Die Datenlänge dieser Instanz beträgt immer Null.
Diese Instanz kann nur im „Consumed Path” (aus Sicht des Slave-Gerätes) benutzt werden.

Instanz 199 ($C7_{hex}$) „Listen Only“

Mit dieser Instanz wird eine Verbindung aufgebaut, die auf einer vorhandenen Exclusive-Owner-Verbindung aufsetzt. Dabei hat die neue Verbindung die gleichen Übertragungsparameter, wie die Exclusive-Owner-Verbindung. Wird die Exclusive-Owner-Verbindung abgebaut, wird auch automatisch diese Verbindung abgebaut. Die Datenlänge dieser Instanz beträgt immer Null.
Diese Instanz kann nur im „Consumed Path” (aus Sicht des Slave-Gerätes) benutzt werden.

Common Services

Tabelle 200: Statische Assembly-Instanzen – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
$0E_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10_{hex}	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

Das Schreiben des Attributes 3 der Assembly-Instanzen 101, 102 und 103 wird von der Software überprüft. Die Überschreitung von Grenzwerten wird festgestellt und, sofern erforderlich, korrigiert. Es wird jedoch keine Schreibanfrage abgelehnt. Das bedeutet, wenn weniger Daten empfangen werden als erwartet, dann werden nur diese Daten geschrieben. Wenn mehr Daten empfangen werden als erwartet, dann werden die empfangenen Daten an der oberen Grenze entfernt. Jedoch wird im Falle von expliziten Nachrichten ein definiertes CIP generiert, obwohl die Daten geschrieben worden sind.

12.3.5.7 Connection (05 hex)

Die Klassen- und Instanz-Attribute dieser Klasse sind nicht sichtbar, da die Verbindungen über den Connection Manager auf- und abgebaut werden.

12.3.5.8 Connection Manager (06 hex)

Das „Connection Manager Object“ stellt die internen Ressourcen bereit, die für die Ein- und Ausgangsdaten und explizite Nachrichten benötigt werden. Weiterhin ist die Verwaltung dieser Ressource eine Aufgabe des „Connection Manager Object“.
Für jede Verbindung (Ein- und Ausgangsdaten oder explizite) wird eine weitere Instanz der Connection-Klasse erzeugt. Die Verbindungsparameter werden dem Dienst „Forward Open“ entnommen, der für den Aufbau einer Verbindung zuständig ist.

Folgende Dienste werden für die erste Instanz unterstützt:

- Forward_Open
- Unconnected_Send
- Forward_Close

Es sind keine Klassen- und Instanz-Attribute sichtbar.

12.3.5.9 Port Class Object (F4 hex)

Das „Port Class Object“ spezifiziert die an dem Feldbuskoppler/-controller vorhandenen CIP-Ports. Für jeden CIP-Port gibt es eine Instanz.

Instanz 0 (Klassenattribute)

Tabelle 201: Port Class (F4 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)
3	Get	Num Instances	UINT	Anzahl von aktuellen Ports	1 (0x0001)
8	Get	Entry Port	UINT	Instanz des Portobjektes, von wo die Anfrage eingetroffen ist	1 (0x0001)
9	Get	All Ports	Array of Struct UINT UINT	Array von Instanz-Attributen 1 und 2 aller Instanzen	0 (0x0000) 0 (0x0000) 4 (0x0004) 2 (0x0002)

Instanz 1

Tabelle 202: Port Class (F4 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Defaultwert
1	Get	V	Port Type	UINT	-	4 (0x0004)
2	Get	V	Port Number	UINT	CIP Portnummer	2 (0x0002) (EtherNet/IP)
3	Get	V	Port Object	UINT	Anzahl von 16 Bit Wörtern im folgenden Pfad	2 (0x0002)
				Padded EPATH	Objekt, das diesen Port verwaltet	0x20 0xF5 0x24 0x01 (entspricht dem TCP/IP-Interface Object)
4	Get	V	Port Name	SHORT_ STRING	Portname	„“
7	Get	V	Node Address	Padded EPATH	Portsegment (IP-Adresse)	Abhängig von der IP-Adresse

Common Services

Tabelle 203: Port Class (F4 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 hex	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.10 TCP/IP Interface Object (F5 $_{hex}$)

Das „TCP/IP Interface Object“ stellt die Einrichtung zur Konfiguration der TCP-IP-Netzwerk-Schnittstelle eines Feldbuskopplers/-controllers bereit.
Beispiele konfigurierbarer Objekte umfassen die IP-Adresse, Netzwerkmaske und Gateway-Adresse des Feldbuskopplers/-controllers.

Bei der zugrunde liegenden physikalischen Kommunikationsschnittstelle, die mit dem TCP/IP-Schnittstellen-Objekt verbunden ist, kann es sich um eine beliebige Schnittstelle handeln, die das TCP/IP-Protokoll unterstützt.
An einem TCP/IP-Schnittstellen-Objekt kann zum Beispiel eine der folgenden Komponenten angeschlossen werden: eine Ethernet-Schnittstelle 802.3, eine ATM-Schnittstelle (Asynchronous Transfer Mode-Schnittstelle) oder eine serielle Schnittstelle für Protokolle wie PPP (Point-to-Point Protocol).
Das TCP/IP-Schnittstellen-Objekt stellt ein Attribut bereit, welches das linkspezifische Objekt für die angeschlossene physikalische Kommunikationsschnittstelle identifiziert. Das linkspezifische Objekt soll üblicherweise linkspezifische Zähler sowie beliebige linkspezifische Konfigurationsattribute bereitstellen.

Jedes Gerät muss genau eine Instanz des TCP/IP-Schnittstellen-Objektes für jede TCP/IP-fähige Kommunikationsschnittstelle unterstützen. Eine Anfrage für den

Zugriff auf die 1. Instanz des TCP/IP-Schnittstellen-Objektes muss sich immer auf die Instanz beziehen, die mit der Schnittstelle verbunden ist, über welche die Anfrage eingegangen ist.

Instanz 0 (Klassenattribute)

Tabelle 204: TCP/IP Interface (F5 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)
3	Get	Num Instances	UINT	Anzahl der aktuell instanzierten Verbindungen	1 (0x0001)

Instanz 1

Tabelle 205: TCP/IP Interface (F5 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Defaultwert
1	Get	V	Status	DWORD	Interface-Status	-
2	Get	V	Configuration Capability	DWORD	Inferface-Flags für mögliche Konfigurationsarten	0x00000017
3	Set	NV	Configuration Control	DWORD	Legt fest wie der Feldbuskoppler/-controller nach dem ersten Neustart zu seiner TCP/IP Konfiguration kommt	0x00000011
4	Get	V	Physical Link Object	STRUCT of		
			Path size	UINT	Anzahl von 16-Bit-Wörtern im folgenden Pfad	0x0002
			Path	Padded EPATH	Logischer Pfad, der auf das physikalische Link-Objekt zeigt	0x20 0xF6 0x24 0x03 (entspricht dem Ethernet Link Object)
5	Set	NV	Interface Configuration	STRUCT of	-	
			IP Address	UDINT	IP-Adresse	0
			Network Mask	UDINT	Netzwerkmaske	0
			Gateway Address	UDINT	IP-Adresse des Standard-Gateway	0
			Name Server	UDINT	IP-Adresse des primären Name-Servers	0
			Name Server 2	UDINT	IP-Adresse des sekundären Name-Servers	0
			Domain Name	STRING	Default-Domain-Name	„“
6	Set	NV	Host Name	STRING	Gerätename	„“

Common Services

Tabelle 206: TCP/IP Interface (F5 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 $_{hex}$	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.11 Ethernet Link Object (F6 $_{hex}$)

Das „Ethernet Link Object“ enthält linkspezifische Zähler- und Statusinformationen für eine Kommunikationsschnittstelle vom Typ Ethernet 802.3. Jedes Gerät muss genau eine Instanz des Ethernet-Link-Objektes für jede Kommunikationsschnittstelle vom Typ Ethernet IEEE 802.3 unterstützen. Für die Geräte kann auch eine Ethernet-Link-Objektinstanz für eine interne Schnittstelle verwendet werden, wie zum Beispiel ein interner Port mit integriertem Switch.

Instanz 0 (Klassenattribute)

Tabelle 207: Ethernet Link (F6 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Version des Objektes	3 (0x0003)
2	Get	Max Instance	UDINT	Max. Anzahl von Instanzen	3 (0x0003)
3	Get	Num Instances	UDINT	Anzahl der aktuell instanzierten Verbindungen	3 (0x0003)

Instanz 1 - Port 1

Tabelle 208: Ethernet Link (F6 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Interface Speed	UDINT	Übertragungs-geschwindigkeit	10 (0x0000000A) oder 100 (0x00000064)
2	Get	Interface Flags	DWORD	Interface Konfigurations-/ Statusinformationen Bit 0: Link-Status Bit 1: Halb-/Vollduplex Bit 2 – 4: Erkennungsstatus Bit 5: Manuelle Ein-stellungen erfordern Reset Bit 6: Lokaler Hardwarefehler Bit 7 – 31: Reserviert	Wert ist von der Ethernet-Verbindung abhängig.
3	Get	Physical Address	ARRAY of 6 UINTs	MAC Layer Address	MAC-ID des Feldbuskopplers/-controllers
6	Set	Interface Control	STRUCT of:	Konfiguration der physikalischen Schnittstelle	-
		Control Bits	WORD	Interface Konfigurations-Bits Bit 0: Automatische Erkennung Bit 1: Vorgabe Duplex-Modus Bit 2 – 15: Reserviert	0x0001
		Forced Interface Speed	UINT	Für die Schnittstelle vorgegebene Geschwindigkeit	10 (0x000A) oder 100 (0x0064)
7	Get	Interface Type	USINT	Schnittstellentyp: Wert 0: Unbekannt Wert 1: Interne Schnittstelle, zum Beispiel bei einem integrierten Switch. Wert 2: Twistet-Pair (z. B. 100Base-TX). Wert 3: Glasfaser (z. B. 100Base-FX). Wert 4 – 256: Reserviert	2 (0x02) – Twisted Pair

Tabelle 208: Ethernet Link (F6 $_{hex}$) – Instanz 1

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
8	Get	Interface Status	USINT	Schnittstellenstatus: Wert 0: Unbekannt Wert 1: Schnittstelle aktiv und zum Senden/Empfangen bereit. Wert 2: Schnittstelle deaktiviert. Wert 3: Schnittstelle testet. Wert 4 – 256: Reserviert	-
9	Get/Set	Admin Status	USINT	Verwaltungsstatus: Wert 0: Reserviert Wert 1: Schnittstelle aktivieren. Wert 2: Schnittstelle deaktivieren. Ist diese die einzige CIP-Schnittstelle, so wird eine Anforderung zum Deaktivieren mit einem Fehler quittiert (Fehlercode 0x09) Wert 3 – 256: Reserviert	1 (0x01)
10	Get	Interface Label	SHORT_ STRING	Name der Schnittstelle	„Port 1“

Instanz 2 - Port 2

Tabelle 209: Ethernet Link (F6 $_{hex}$) – Instanz 2

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Interface Speed	UDINT	Übertragungs-geschwindigkeit	10 (0x0000000A) oder 100 (0x00000064)
2	Get	Interface Flags	DWORD	Interface Konfigurations-/ Statusinformationen Bit 0: Link-Status Bit 1: Halb-/Vollduplex Bit 2 – 4: Erkennungsstatus Bit 5: Manuelle Ein-stellungen erfordern Reset Bit 6: Lokaler Hardwarefehler Bit 7 – 31: Reserviert	Wert ist von der Ethernet-Verbindung abhängig.
3	Get	Physical Address	ARRAY of 6 UINTs	MAC Layer Address	MAC-ID des Feldbuskopplers/-controllers
6	Set	Interface Control	STRUCT of:	Konfiguration der physikalischen Schnittstelle	-
		Control Bits	WORD	Interface Konfigurations-Bits Bit 0: Automatische Erkennung Bit 1: Vorgabe Duplex-Modus Bit 2 – 15: Reserviert	0x0001
		Forced Interface Speed	UINT	Für die Schnittstelle vorgegebene Geschwindigkeit	10 (0x000A) oder 100 (0x0064)
7	Get	Interface Type	USINT	Schnittstellentyp: Wert 0: Unbekannt Wert 1: Interne Schnittstelle, zum Beispiel bei einem integrierten Switch. Wert 2: Twistet-Pair (z. B. 100Base-TX). Wert 3: Glasfaser (z. B. 100Base-FX). Wert 4 – 256: Reserviert	2 (0x02) – Twisted Pair

Tabelle 209: Ethernet Link ($F6_{hex}$) – Instanz 2

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
8	Get	Interface Status	USINT	Schnittstellenstatus: Wert 0: Unbekannt Wert 1: Schnittstelle aktiv und zum Senden/Empfangen bereit. Wert 2: Schnittstelle deaktiviert. Wert 3: Schnittstelle testet. Wert 4 – 256: Reserviert	-
9	Get/Set	Admin Status	USINT	Verwaltungsstatus: Wert 0: Reserviert Wert 1: Schnittstelle aktivieren. Wert 2: Schnittstelle deaktivieren. Ist diese die einzige CIP-Schnittstelle, so wird eine Anforderung zum Deaktivieren mit einem Fehler quittiert (Fehlercode 0x09) Wert 3 – 256: Reserviert	1 (0x01)
10	Get	Interface Label	SHORT_ STRING	Name der Schnittstelle	„Port 2“

Instanz 3 – Interner Port 3

Tabelle 210: Ethernet Link ($F6_{hex}$) – Instanz 3

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Interface Speed	UDINT	Übertragungs-geschwindigkeit	100 (0x00000064)
2	Get	Interface Flags	DWORD	Interface Konfigurations-/ Statusinformationen Bit 0: Link-Status Bit 1: Halb-/Vollduplex Bit 2 – 4: Erkennungsstatus Bit 5: Manuelle Ein-stellungen erfordern Reset Bit 6: Lokaler Hardwarefehler Bit 7 – 31: Reserviert	Wert ist von der Ethernet-Verbindung abhängig.
3	Get	Physical Address	ARRAY of 6 UINTs	MAC Layer Address	MAC-ID des Feldbuskopplers/-controllers
6	Get	Interface Control	STRUCT of:	Konfiguration der physikalischen Schnittstelle	-
		Control Bits	WORD	Interface Konfigurations-Bits Bit 0: Automatische Erkennung Bit 1: Vorgabe Duplex-Modus Bit 2 – 15: Reserviert	0x0002
		Forced Interface Speed	UINT	Für die Schnittstelle vorgegebene Geschwindigkeit	100 (0x0064)
7	Get	Interface Type	USINT	Schnittstellentyp: Wert 0: Unbekannt Wert 1: Interne Schnittstelle, zum Beispiel bei einem integrierten Switch. Wert 2: Twistet-Pair (z. B. 100Base-TX). Wert 3: Glasfaser (z. B. 100Base-FX). Wert 4 – 256: Reserviert	1 (0x01)

Tabelle 210: Ethernet Link (F6 $_{hex}$) – Instanz 3

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
8	Get	Interface Status	USINT	Schnittstellenstatus: Wert 0: Unbekannt Wert 1: Schnittstelle aktiv und zum Senden/Empfangen bereit. Wert 2: Schnittstelle deaktiviert. Wert 3: Schnittstelle testet. Wert 4 – 256: Reserviert	-
9	Get	Admin Status	USINT	Verwaltungsstatus: Wert 0: Reserviert Wert 1: Schnittstelle aktivieren. Wert 2: Schnittstelle deaktivieren. Ist diese die einzige CIP-Schnittstelle, so wird eine Anforderung zum Deaktivieren mit einem Fehler quittiert (Fehlercode 0x09) Wert 3 – 256: Reserviert	1 (0x01)
10	Get	Interface Label	SHORT_ STRING	Name der Schnittstelle	„Internal Port 3“

Common Services

Tabelle 211: Ethernet Link (F6 $_{hex}$) – Common Services

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
01 hex	Ja	Ja	Get_Attribute_All	Liefert den Inhalt aller Attribute
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

Hinweis

Änderungen mittels „Set_Attribute_Single“ nicht sofort wirksam!
Attribute (speziell Attribut 6 und 9), die Sie über den Service „Set_Attribute_Single“ verändern, werden erst nach dem nächsten Neustart des Feldbuskopplers/-controllers wirksam.

12.3.5.12 Coupler/Controller Configuration Object (64 hex)

Die Konfigurationsklasse des Feldbuskopplers/-controllers ermöglicht das Lesen und Konfigurieren einiger wichtiger Prozessparameter des Feldbusses. Die folgende Auflistung erklärt ausführlich alle unterstützten Instanzen und Attribute.

Instanz 0 (Klassenattribute)

Tabelle 212: Coupler/Controller Configuration (64 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	1 (0x0001)

Instanz 1

Tabelle 213: Coupler/Controller Configuration (64 hex) – Instanz 1

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
5 (0x05)	Get	V	ProcessState	USINT	Koppler-/Controller-Status, Fehlermaske: Bit 0: Klemmenbusfehler Bit 3: Klemmendiagnose (0x08) Bit 7: Feldbusfehler (0x80)	0
6 (0x06)	Get	V	DNS_i_ Trmnldia	UINT	Klemmendiagnose: Bit 0..7: Klemmennummer Bit 8..14: Klemmenkanal Bit 15: 0/1 Fehler behoben/aufgetreten	0
7 (0x07)	Get	V	CnfLen. AnalogOut	UINT	Anzahl I/O-Bits für die analogen Ausgänge	-
8 (0x08)	Get	V	CnfLen. AnalogInp	UINT	Anzahl I/O-Bits für die analogen Eingänge	-
9 (0x09)	Get	V	CnfLen. DigitalOut	UINT	Anzahl I/O-Bits für die digitalen Ausgänge	-
10 (0x0A)	Get	V	CnfLen. DigitalInp	UINT	Anzahl I/O-Bits für die digitalen Eingänge	-
11 (0x0B)	Set	NV	Bk_Fault_ Reaction	USINT	Feldbusfehlerreaktion 0: stoppt lokale I/O-Zyklen 1: alle Ausgänge zu 0 setzen 2: keine Fehlerreaktion 3: keine Fehlerreaktion 4: PFC-Task übernimmt die Kontrolle der Ausgänge (gilt für Controller)	1
12..26 (0x0C...0x1A)	Reserviert aus Kompatibilität zu DeviceNet					
40..43 (0x28...0x2B)	Reserviert aus Kompatibilität zu DeviceNet					
45 (0x2D)	Get	V	Bk_Led_Err_ Code	UINT	I/O-LED Error-Code	0
46 (0x2E)	Get	V	Bk_Led_Err_ Arg	UINT	I/O-LED Error-Argument	0

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
47 (0x2F)	Get	V	Bk_Diag_Value	UINT	Enthält das Diagnosebyte. **Achtung:** Dieses Attribut muss vor dem Attribut 6 (DNS_i_Trmnldia) gelesen werden, da mit dem Lesen von Attribut 6 das Diagnose-byte von der nächsten Diagnose anliegt.	0
100 (0x64)	Set	NV	Bk_FbInp_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Eingangs-variablen, die zu dem Assembly-Objekt hinzugefügt werden. Diese Anzahl wird zu dem konsumierenden Pfad hinzugezählt. Assembly-Instanzen 101...103	0
101 (0x65)	Set	NV	Bk_FbOut_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Ausgangs-variablen, die zu dem Assembly-Objekt hinzugefügt werden. Diese Anzahl wird zu dem produzierenden Pfad hinzugezählt. Assembly-Instanzen 104...109	0
102 (0x66)	Set	NV	Bk_FbInp_Plc Only_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Eingangs-variablen, die mittels Assembly-Instanz 111 empfangen werden	4
103 (0x67)	Set	NV	Bk_FbInp_Start Plc_Var_Cnt	UINT	Legt fest, ab welcher Position die PFC-Eingangsvariablen für die Assembly-Instanz 111 empfangen werden	0
104 (0x68)	Set	NV	Bk_FbOut_Plc Only_Var_Cnt	UINT	Bestimmt die Anzahl der Bytes für die PFC-Ausgangs-variablen die mittels Assembly-Instanz 110 übertragen werden	4
105 (0x69)	Set	NV	Bk_FbOut_Star t Plc_Var_Cnt	UINT	Legt fest, ab welcher Position die PFC-Ausgangsvariablen für die Assembly-Instanz 110 übertragen werden	0

Attribut ID	Zugriff	NV	Name	Datentyp	Beschreibung	Default-wert
120 (0x78)	Set	NV	Bk_Header CfgOT	UINT	Gibt an, ob der RUN/IDLE-Header benutzt wird Originator → Target Richtung 0: wird verwendet 1: wird nicht verwendet	0x0000
121(0x79)	Set	NV	Bk_Header CfgTO	UINT	Gibt an, ob der RUN/IDLE-Header benutzt wird Target → Originator Richtung 0: wird verwendet 1: wird nicht verwendet	0x0001

Common Service

Tabelle 214: Coupler/Controller Configuration (64 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.13 Discrete Input Point (65 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten digitalen Eingangspunktes.

Instanz 0 (Klassenattribute)

Tabelle 215: Discrete Input Point (65 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. digitaler Eingangswert)

Tabelle 216: Discrete Input Point (65 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 217: Discrete Input Point (65 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.14 Discrete Input Point Extended 1 (69 hex)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 255 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse „Discrete Input Point Extended 1“ deckt die DIPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 218: Discrete Input Point Extended 1(69 hex,) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. digitaler Eingangswert)

Tabelle 219: Discrete Input Point Extended 1 (69 hex,) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 220: Discrete Input Point Extended 1 (69 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.15 Discrete Input Point Extended 2 (6D hex)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 510 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse „Discrete Input Point Extended 2“ deckt die DIPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 221: Discrete Input Point Extended 2 (6D hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. digitaler Eingangswert)

Tabelle 222: Discrete Input Point Extended 2 (6D $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 223: Discrete Input Point Extended 2 (6D $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.16 Discrete Input Point Extended 3 (71 hex)

Die Erweiterung der Klasse „Discrete Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 765 digitale Eingangspunkte (DIPs) enthält. Der Instanzbereich der Klasse „Discrete Input Point Extended 3“ deckt die DIPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 224: Discrete Input Point Extended 3 (71 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. digitaler Eingangswert)

Tabelle 225: Discrete Input Point Extended 3 (71 hex) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DipObj_Value	BYTE	Digitaler Eingang (nur Bit 0 gültig)	-

Common Services

Tabelle 226: Discrete Input Point Extended 3 (71 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.17 Discrete Output Point (66 hex)

Diese Klasse ermöglicht den Austausch von Daten für einen bestimmten digitalen Ausgangspunkt.

Instanz 0 (Klassenattribute)

Tabelle 227: Discrete Output Point (66 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. digitaler Ausgangswert)

Tabelle 228: Discrete Output Point (66 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 229: Discrete Output Point (66 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.18 Discrete Output Point Extended 1 (6A $_{hex}$)

Die Erweiterung der Klasse „Discrete Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 255 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 1“ deckt die DOPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 230: Discrete Output Point Extended 1 (6A $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. digitaler Ausgangswert)

Tabelle 231: Discrete Output Point Extended 1 (6A $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 232: Discrete Output Point Extended 1 (6A $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.19 Discrete Output Point Extended 2 (6E $_{hex}$)

Die Erweiterung der Klasse „Discrete Output Point" ermöglicht den Austausch von Daten eines Feldbusknotens, der über 510 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 2" deckt die DOPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 233: Discrete Output Point Extended 2 (6E $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. digitaler Ausgangswert)

Tabelle 234: Discrete Output Point Extended 2 (6E $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 235: Discrete Output Point Extended 2 (6E $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.20 Discrete Output Point Extended 3 (72 hex)

Die Erweiterung der Klasse „Discrete Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 765 digitale Ausgangspunkte (DOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 2“ deckt die DOPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 236: Discrete Output Point Extended 3 (72 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. digitaler Ausgangswert)

Tabelle 237: Discrete Output Point Extended 3 (72 hex) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	DopObj_Value	BYTE	Digitaler Ausgang (nur Bit 0 gültig)	-

Common Services

Tabelle 238: Discrete Output Point Extended 2 (6E hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.21 Analog Input Point (67 hex)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten analogen Eingangspunktes (AIP). Ein analoger Eingangspunkt ist ein Teil eines analogen Eingangsmoduls.

Instanz 0 (Klassenattribute)

Tabelle 239: Analog Input Point (67 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. analoger Eingangswert)

Tabelle 240: Analog Input Point (67 hex) – Instanz 1 ... 255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 241: Analog Input Point (67 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.22 Analog Input Point Extended 1 (6B $_{hex}$)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 255 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse „Analog Input Point Extended 1“ deckt die AIPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 242: Analog Input Point Extended 1 (6B $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. analoger Eingangswert)

Tabelle 243: Analog Input Point Extended 1 (6B $_{hex}$) – Instanz 256 ... 510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 244: Analog Input Point Extended 1 (6B $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.23 Analog Input Point Extended 2 (6F $_{hex}$)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 510 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse „Analog Input Point Extended 2“ deckt die AIPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 245: Analog Input Point Extended 2 (6F $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. analoger Eingangswert)

Tabelle 246: Analog Input Point Extended 2 (6F $_{hex}$) – Instanz 511 ... 765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 247: Analog Input Point Extended 2 (6F $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.24 Analog Input Point Extended 3 (73 hex)

Die Erweiterung der Klasse „Analog Input Point“ ermöglicht das Lesen von Daten eines Feldbusknotens, der über 765 analoge Ausgänge (AIPs) enthält. Der Instanzbereich der Klasse „Analog Input Point Extended 3“ deckt die AIPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 248: Analog Input Point Extended 3 (73 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. analoger Eingangswert)

Tabelle 249: Analog Input Point Extended 3 (73 hex) – Instanz 766 ... 1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AipObj_Value	ARRAY of BYTE	Analoger Eingang	-
2	Get	AipObj_Value_ Length	USINT	Länge der Eingangsdaten AipObj_Value (in Byte)	-

Common Services

Tabelle 250: Analog Input Point Extended 3 (73 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.25 Analog Output Point (68 hex)

Diese Klasse ermöglicht das Lesen von Daten eines bestimmten analogen Ausgangspunktes (AOP). Ein analoger Ausgangspunkt ist ein Teil eines analogen Ausgangsmoduls.

Instanz 0 (Klassenattribute)

Tabelle 251: Analog Output Point (68 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1 ... 255 (1. bis 255. Analoger Ausgangswert)

Tabelle 252: Analog Output Point (68 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 253: Analog Output Point (68 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.26 Analog Output Point Extended 1 (6C $_{hex}$)

Die Erweiterung der Klasse „Analog Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 255 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 1“ deckt die AOPs von 256 bis 510 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 254: Analog Output Point Extended 1 (6C $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 ... 510 (256. bis 510. Analoger Ausgangswert)

Tabelle 255: Analog Output Point Extended 1 (6C $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 256: Analog Output Point Extended 1 (6C $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.27 Analog Output Point Extended 2 (70 $_{hex}$)

Die Erweiterung der Klasse „Analog Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 510 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 2“ deckt die AOPs von 511 bis 765 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 257: Analog Output Point Extended 2 (70 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 511 ... 765 (511. bis 765. Analoger Ausgangswert)

Tabelle 258: Analog Output Point Extended 2 (70 $_{hex}$) – Instanz 511...765

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 259: Analog Output Point Extended 2 (70 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.28 Analog Output Point Extended 3 (74 hex)

Die Erweiterung der Klasse „Analog Output Point“ ermöglicht den Austausch von Daten eines Feldbusknotens, der über 765 analoge Ausgangspunkte (AOPs) enthält. Der Instanzbereich der Klasse „Discrete Output Point Extended 3“ deckt die AOPs von 766 bis 1020 in dem Feldbusknoten ab.

Instanz 0 (Klassenattribute)

Tabelle 260: Analog Output Point Extended 3 (74 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 766 ... 1020 (766. bis 1020. Analoger Ausgangswert)

Tabelle 261: Analog Output Point Extended 3 (74 hex) – Instanz 766...1020

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	AopObj_Value	ARRAY of BYTE	Analoger Ausgang	-
2	Get	AopObj_Value _Length	USINT	Länge der Ausgangsdaten AopObj_Value (in Byte)	-

Common Services

Tabelle 262: Analog Output Point Extended 3 (74 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.29 Module Configuration (80 hex)

Instanz 0 (Klassenattribute)

Tabelle 263: Module Configuration (80 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 1..255 (0. bis 254. Busklemme)

Tabelle 264: Module Configuration (80 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Default-wert
1	Get	ModulDescription	WORD	Beschreibung angeschlossener Busklemmen (Busklemme 0 = Feldbuskoppler/-controller) Bit 0: Busklemme hat Eingänge Bit 1: Busklemme hat Ausgänge Bit 8-14: Datenbreite intern in Bit Bit 15: 0/1 Analoge/Digitale Busklemme Bei analogen Busklemmen bezeichnen die Bits 0-14 den Busklemmentyp, z. B. 401 für die Busklemme 750-401	-

Common Services

Tabelle 265: Module Configuration (80 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.30 Module Configuration Extended (81 hex)

Wie „Module Configuration (80 hex)“, jedoch enthält diese Klasse nur die Beschreibung von Busklemme 255.

Instanz 0 (Klassenattribute)

Tabelle 266: Module Configuration Extended (81 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl von Instanzen	-

Instanz 256 (255. Busklemme)

Tabelle 267: Module Configuration Extended (81 hex) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Default-wert
1	Get	ModulDescription	WORD	Beschreibung angeschlossener Busklemmen (Busklemme 0 = Feldbuskoppler/-controller) Bit 0: Busklemme hat Eingänge Bit 1: Busklemme hat Ausgänge Bit 8-14: Datenbreite intern in Bit Bit 15: 0/1 Analoge/Digitale Busklemme Bei analogen Busklemmen bezeichnen die Bits 0-14 den Busklemmentyp, z. B. 401 für die Busklemme 750-401	-

Common Services

Tabelle 268: Module Configuration Extended (81 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.31 Input Fieldbus Variable USINT (A0 $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %IB2552...%IB2806.

Instanz 0 (Klassenattribute)

Tabelle 269: Input Fieldbus Variable USINT (A0 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Eingangsvariable)

Tabelle 270: Input Fieldbus Variable USINT (A0 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 271: Input fieldbus variable USINT (A0 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.32 Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$)

Die Erweiterung der Klasse „Input Fieldbus Variable USINT“ ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse “Input Fieldbus Variable USINT Extended 1“ deckt die SPS-Eingangsvariablendaten von 256 bis 510 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %IB2807...%IB3061.

Instanz 0 (Klassenattribute)

Tabelle 272: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 256...510 (256. bis 510. Eingangsvariable)

Tabelle 273: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 274: Input fieldbus variable USINT Extended 1 (A1 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.33 Input Fieldbus Variable USINT Extended 2 (A2 $_{hex}$)

Die Erweiterung der Klasse „Input Fieldbus Variable USINT“ ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse “Input Fieldbus Variable USINT Extended 2“ deckt die SPS-Eingangsvariablendaten von 511 bis 512 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %IB3062...%IB3063.

Instanz 0 (Klassenattribute)

Tabelle 275: Input Fieldbus Variable USINT Extended 2 (A2 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	2 (0x0002)

Instanz 511...512 (511. bis 512. Eingangsvariable)

Tabelle 276: Input Fieldbus Variable USINT Extended 2 (A2 $_{hex}$) – Instanz 511...512

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	USINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 277: Input fieldbus variable USINT Extended 2 (A2 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.34 Output Fieldbus Variable USINT (A3 $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariablen %QB2552...%QB2806.

Instanz 0 (Klassenattribute)

Tabelle 278: Output Fieldbus Variable USINT (A3 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Ausgangsvariable)

Tabelle 279: Output Fieldbus Variable USINT (A3 $_{hex}$) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 280: Output Fieldbus Variable USINT (A3 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.35 Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$)

Die Erweiterung der Klasse „Output Fieldbus Variable USINT“ ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse “Output Fieldbus Variable USINT Extended 1“ deckt die SPS-Ausgangsvariablendaten von 256 bis 510 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariablen %QB2807...%QB3061.

Instanz 0 (Klassenattribute)

Tabelle 281: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 256...510 (256. bis 510. Ausgangsvariable)

Tabelle 282: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Instanz 256...510

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 283: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.36 Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$)

Die Erweiterung der Klasse „Output Fieldbus Variable USINT“ ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse “Output Fieldbus Variable USINT Extended 2“ deckt die SPS-Ausgangsvariablendaten von 511 bis 512 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariablen %QB3062...%QB3063.

Instanz 0 (Klassenattribute)

Tabelle 284: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	2 (0x0002)

Instanz 511...512 (511. bis 512. Ausgangsvariable)

Tabelle 285: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Instanz 511...512

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	USINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 286: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.37 Input Fieldbus Variable UINT (A6 hex)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %IW1276...%IW1530.

Instanz 0 (Klassenattribute)

Tabelle 287: Input Fieldbus Variable UINT (A6 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Eingangsvariable)

Tabelle 288: Input Fieldbus Variable UINT (A6 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 289: Input Fieldbus Variable UINT (A6 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 hex	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.38 Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$)

Die Erweiterung der Klasse „Input Fieldbus Variable UINT“ ermöglicht das Lesen von SPS-Eingangsvariablendaten. Der Instanzbereich der Klasse “Input Fieldbus Variable UINT Extended 1“ deckt die SPS-Eingangsvariablendaten von der SPS-Eingangsvariablen 256 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS der SPS-Adresse für die Eingangsvariable %IW1531.

Instanz 0 (Klassenattribute)

Tabelle 290: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	1 (0x0001)

Instanz 256 (256. Eingangsvariable)

Tabelle 291: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 292: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.39 Output Fieldbus Variable UINT (A8 hex)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariablen %QW1276...%QW1530.

Instanz 0 (Klassenattribute)

Tabelle 293: Output Fieldbus Variable UINT (A8 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	255 (0x00FF)

Instanz 1...255 (1. bis 255. Ausgangsvariable)

Tabelle 294: Output Fieldbus Variable UINT (A8 hex) – Instanz 1...255

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	UINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 295: Output Fieldbus Variable UINT (A8 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.40 Output Fieldbus Variable UINT Extended 1 (A9 hex)

Die Erweiterung der Klasse „Output Fieldbus Variable UINT“ ermöglicht den Austausch von SPS-Ausgangsvariablendaten. Der Instanzbereich der Klasse “Output Fieldbus Variable UINT Extended 1“ deckt die SPS-Ausgangsvariablendaten von der SPS-Ausgangsvariablen 256 ab.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariable %QW1531.

Instanz 0 (Klassenattribute)

Tabelle 296: Output Fieldbus Variable UINT Extended 1 (A9 hex) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	1 (0x0001)

Instanz 256 (256. Ausgangsvariable)

Tabelle 297: Output Fieldbus Variable UINT Extended 1 (A9 hex) – Instanz 256

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Fb_Out_Var	UINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 298: Output Fieldbus Variable UINT Extended 1 (A9 hex) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E hex	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.41 Input Fieldbus Variable UDINT (AA $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %ID638 ... %ID765.

Instanz 0 (Klassenattribute)

Tabelle 299: Input Fieldbus Variable UDINT (AA $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Eingangsvariable)

Tabelle 300: Input Fieldbus Variable UDINT (AA $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UDINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 301: Input Fieldbus Variable UDINT (AA $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.42 Input Fieldbus Variable UDINT Offset (AB $_{hex}$)

Diese Klasse ermöglicht das Lesen von Daten einer bestimmten SPS-Eingangsvariablen.
Mit einem Offset von 2 Byte auf die Adressen der Klasse “Input Fieldbus Variable UDINT (AA $_{hex}$)“ entspricht dieses bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Eingangsvariablen %ID638 ... %ID765.

Information

Informationen zur Verwendung des Offsets
Unter „Offset von 2 Byte“ ist Folgendes zu verstehen:
Wird Instanz 1 dieser Klasse ausgelesen, so erhalten Sie das High-Word der Adresse %ID638 und das Low-Word der Adresse %ID639, usw.
Wird Instanz 128 ausgelesen, so erhalten Sie nur das High-Word der Adresse %ID765.

Instanz 0 (Klassenattribute)

Tabelle 302: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x080)

Instanz 1...128 (1. bis 128. Eingangsvariable)

Tabelle 303: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_In_Var	UDINT	Feldbus-Eingangsvariable der SPS	0

Common Services

Tabelle 304: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes
10 $_{hex}$	Nein	Ja	Set_Attribute_Single	Modifiziert einen Attribut-Wert

12.3.5.43 Output Fieldbus Variable UDINT (AC $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Dieses entspricht bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die Ausgangsvariablen %QD638 ... %QD765.

Instanz 0 (Klassenattribute)

Tabelle 305: Output Fieldbus Variable UDINT (AC $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Ausgangsvariable)

Tabelle 306: Output Fieldbus Variable UDINT (AC $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_Out_Var	UDINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 307: Output Fieldbus Variable UDINT (AC $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

12.3.5.44 Output Fieldbus Variable UDINT Offset (AD $_{hex}$)

Diese Klasse ermöglicht den Austausch von Daten einer bestimmten SPS-Ausgangsvariablen.
Mit einem Offset von 2 Byte auf die Adressen der Klasse “Output Fieldbus Variable UDINT (AC $_{hex}$)“ entspricht dieses bei WAGO-I/O-*PRO* bzw. CODESYS den SPS-Adressen für die %QD638 ... %QD765.

Information

Informationen zur Verwendung des Offsets
Unter „Offset von 2 Byte“ ist Folgendes zu verstehen:
Wird Instanz 1 dieser Klasse ausgelesen, so erhalten Sie das High-Word der Adresse %ID638 und das Low-Word der Adresse %ID639, usw.
Wird Instanz 128 ausgelesen, so erhalten Sie nur das High-Word der Adresse %ID765.

Instanz 0 (Klassenattribute)

Tabelle 308: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Klasse

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Get	Revision	UINT	Revision dieses Objektes	1 (0x0001)
2	Get	Max Instance	UINT	Max. Anzahl an Instanzen	128 (0x0080)

Instanz 1...128 (1. bis 128. Ausgangsvariable)

Tabelle 309: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Instanz 1...128

Attribut ID	Zugriff	Name	Datentyp	Beschreibung	Defaultwert
1	Set	Fb_Out_Var	UDINT	Feldbus-Ausgangsvariable der SPS	0

Common Services

Tabelle 310: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Common service

Servicecode	Service vorhanden		Service-Name	Beschreibung
	Klasse	Instanz		
0E $_{hex}$	Ja	Ja	Get_Attribute_Single	Liefert den Inhalt des entsprechenden Attributes

13 Busklemmen

13.1 Übersicht

Für den Aufbau von Applikationen mit dem WAGO-I/O-SYSTEM 750/753 sind verschiedene Arten von Busklemmen verfügbar:

- Digitaleingangsklemmen
- Digitalausgangsklemmen
- Analogeingangsklemmen
- Analogausgangsklemmen
- Sonderklemmen
- Systemklemmen

Eine detaillierte Beschreibung zu jeder Busklemme und deren Varianten entnehmen Sie den Handbüchern zu den Busklemmen.

Sie finden diese Handbücher auf der Internetseite www.wago.com.

Information

Weitere Information zum WAGO-I/O-SYSTEM
Aktuelle Informationen zum modularen WAGO-I/O-SYSTEM finden Sie auf der Internetseite unter: www.wago.com.

13.2 Aufbau der Prozessdaten für MODBUS/TCP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei MODBUS/TCP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Im Folgenden wird für alle Busklemmen des WAGO-I/O-SYSTEMs 750 und 753 die feldbusspezifische Darstellung im Prozessabbild für MODBUS/TCP beschrieben und der Aufbau der Prozesswerte gezeigt.

ACHTUNG

Geräteschäden durch falsche Adressierung!
Zur Vermeidung von Geräteschäden im Feldbereich, müssen Sie bei der Adressierung einer an beliebiger Position im Feldbusknoten befindlichen Busklemme, die Prozessdaten aller vorherigen byte- bzw. bitweise-orientierten Busklemmen berücksichtigen.

Für das PFC-Prozessabbild des Feldbuscontrollers ist der Aufbau der Prozesswerte identisch.

13.2.1 Digitaleingangsklemmen

Die Digitaleingangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Signalzustand des jeweiligen Kanals angibt. Diese Bits werden in das Eingangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie z. B. Drahtbruch und/oder Kurzschluss.

Sofern in dem Knoten auch Analogeingangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Eingangsdaten in dem Eingangsprozessabbild angehängt.

13.2.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose

750-435

Tabelle 311: 1-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 1	Datenbit DI 1

13.2.1.2 2-Kanal-Digitaleingangsklemmen

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (und alle Varianten),
753-400, -401, -405, -406, -410, -411, -412, -427

Tabelle 312: 2-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose

750-419, -421, -424, -425
753-421, -424, -425

Tabelle 313: 2-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

750-418
753-418

Die Digitaleingangsklemme liefert über die Prozesswerte im Eingangsprozessabbild hinaus 4 Bit Daten, die im Ausgangsprozessabbild dargestellt werden.

Tabelle 314: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Quittierungsbit Q 2 Kanal 2	Quittierungsbit Q 1 Kanal 1	0	0

13.2.1.5 4-Kanal-Digitaleingangsklemmen

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422, -1423
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Tabelle 315: 4-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.6 8-Kanal-Digitaleingangsklemmen

750-430, -431, -436, -437, -1415, -1416, -1417, -1418
753-430, -431, -434

Tabelle 316: 8-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.2.1.7 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten

750-1425

Die Digitaleingangsklemme PTC liefert über einen logischen Kanal 2 Byte für das Ein- und Ausgangsprozessabbild.

Der Signalzustand der PTC-Eingänge DI1 … DI8 wird über das Eingangsdatenbyte D0 an den Feldbuskoppler/-controller übertragen.
Die Fehlerzustände werden über das Eingangsdatenbyte D1 übertragen.

Über das Ausgangsdatenbyte D1 werden die Kanäle 1 … 8 ein- oder ausgeschaltet. Das Ausgangsdatenbyte D0 ist reserviert und hat immer den Wert „0“.

Tabelle 317: 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten

Eingangsprozessabbild															
Eingangsbyte D0								**Eingangsbyte D1**							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Signal-zu-stand DI 8 Kanal 8	Signal-zu-stand DI 7 Kanal 7	Signal-zu-stand DI 6 Kanal 6	Signal-zu-stand DI 5 Kanal 5	Signal-zu-stand DI 4 Kanal 4	Signal-zu-stand DI 3 Kanal 3	Signal-zu-stand DI 2 Kanal 2	Signal-zu-stand DI 1 Kanal 1	Draht-bruch/ Kurz-schluss DB/KS 8 Kanal 8	Draht-bruch/ Kurz-schluss DB/KS 7 Kanal 7	Draht-bruch/ Kurz-schluss DB/KS 6 Kanal 6	Draht-bruch/ Kurz-schluss DB/KS 5 Kanal 5	Draht-bruch/ Kurz-schluss DB/KS 4 Kanal 4	Draht-bruch/ Kurz-schluss DB/KS 3 Kanal 3	Draht-bruch/ Kurz-schluss DB/KS 2 Kanal 2	Draht-bruch/ Kurz-schluss DB/KS 1 Kanal 1

Ausgangsprozessabbild															
Ausgangsbyte D0								**Ausgangsbyte D1**							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	0	0	0	0	0	0	0	DI Off 8 Kanal 8 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 7 Kanal 7 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 6 Kanal 6 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 5 Kanal 5 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 4 Kanal 4 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 3 Kanal 3 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 2 Kanal 2 0: Kanal einge-chaltet 1: Kanal ausge-schaltet	DI Off 1 Kanal 1 0: Kanal einge-chaltet 1: Kanal ausge-schaltet

13.2.1.8 16-Kanal-Digitaleingangsklemmen

750-1400, -1402, -1405, -1406, -1407

Tabelle 318: 16-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Daten bit DI 16 Kanal 16	Daten bit DI 15 Kanal 15	Daten bit DI 14 Kanal 14	Daten bit DI 13 Kanal 13	Daten bit DI 12 Kanal 12	Daten bit DI 11 Kanal 11	Daten bit DI 10 Kanal 10	Daten bit DI 9 Kanal 9	Daten bit DI 8 Kanal 8	Daten bit DI 7 Kanal 7	Daten bit DI 6 Kanal 6	Daten bit DI 5 Kanal 5	Daten bit DI 4 Kanal 4	Daten bit DI 3 Kanal 3	Daten bit DI 2 Kanal 2	Daten bit DI 1 Kanal 1

13.2.2 Digitalausgangsklemmen

Die Digitalausgangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Status des jeweiligen Kanals angibt. Diese Bits werden in das Ausgangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie Drahtbruch und/oder Kurzschluss. Bei einigen Busklemmen müssen, bei gesetztem Diagnosebit, zusätzlich die Datenbits ausgewertet werden.

Sofern in dem Knoten auch Analogausgangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Ausgangsdaten in dem Ausgangsprozessabbild angehängt.

13.2.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

750-523

Die Digitalausgangsklemmen liefern über das eine Prozesswert-Bit im Ausgangsprozessabbild hinaus 1 Bit, das im Eingangsprozessabbild dargestellt wird. Dieses Statusbit zeigt den „Handbetrieb“ an.

Tabelle 319: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	Statusbit “Hand-betrieb“

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	steuert DO 1 Kanal 1

13.2.2.2 2-Kanal-Digitalausgangsklemmen

750-501, -502, -509, -512, -513, -514, -517, -535, (und alle Varianten),
753-501, -502, -509, -512, -513, -514, -517

Tabelle 320: 2-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-507 (-508), -522,
753-507

Die Digitalausgangsklemmen liefern über die 2-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 2 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 321: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

750-506,
753-506

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die durch einen 2-Bit-Fehlercode eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 322: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 3 Kanal 2	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Diagnosebit S 0 Kanal 1

Diagnosebits S1/S0, S3/S2: = ‘00’ normaler Betrieb
Diagnosebits S1/S0, S3/S2: = ‘01’ keine Last angeschlossen/Kurzschluss gegen +24 V
Diagnosebits S1/S0, S3/S2: = ‘10’ Kurzschluss gegen GND/Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				nicht genutzt	nicht genutzt	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.4 4-Kanal-Digitalausgangsklemmen

750-504, -516, -519, -531,
753-504, -516, -531, -540

Tabelle 323: 4-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-532

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 324: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = ‘0’ kein Fehler
Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.6 8-Kanal-Digitalausgangsklemmen

750-530, -536, -1515, -1516
753-530, -534

Tabelle 325: 8-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-537

Die Digitalausgangsklemmen liefern über die 8-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 8 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 326: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnosebit S 8 Kanal 8	Diagnosebit S 7 Kanal 7	Diagnosebit S 6 Kanal 6	Diagnosebit S 5 Kanal 5	Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Diagnosebit S = ‘0’ kein Fehler
Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.8 16-Kanal-Digitalausgangsklemmen

750-1500, -1501, -1504, -1505

Tabelle 327: 16-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 16 Kanal 16	steuert DO 15 Kanal 15	steuert DO 14 Kanal 14	steuert DO 13 Kanal 13	steuert DO 12 Kanal 12	steuert DO 11 Kanal 11	steuert DO 10 Kanal 10	steuert DO 9 Kanal 9	steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.2.9 8-Kanal-Digitaleingangsklemmen/-Digitalausgangsklemmen

750-1502, -1506

Tabelle 328: 8-Kanal-Digitalein-/ -ausgangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

13.2.3 Analogeingangsklemmen

Die Analogeingangsklemmen liefern je Kanal 16-Bit-Messwerte und 8 Steuer-/ Statusbits.

MODBUS/TCP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Eingangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/ -controller mit MODBUS/TCP deshalb nur die 16-Bit-Messwerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitaleingangsklemmen gesteckt sind, werden die analogen Eingangsdaten immer vor die digitalen Daten in das Eingangs-prozessabbild abgebildet.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: www.wago.com.

13.2.3.1 1-Kanal-Analogeingangsklemmen

750-491, (und alle Varianten)

Tabelle 329: 1-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert U_D
1	D3	D2	Messwert U_{ref}

13.2.3.2 2-Kanal-Analogeingangsklemmen

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (und alle Varianten),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (und alle Varianten)

Tabelle 330: 2-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2

13.2.3.3 4-Kanal-Analogeingangsklemmen

750-450, -453, -455, -457, -459, -460, -468, (und alle Varianten),
753-453, -455, -457, -459

Tabelle 331: 4-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2
2	D5	D4	Messwert Kanal 3
3	D7	D6	Messwert Kanal 4

13.2.3.4 3-Phasen-Leistungsmessklemme

750-493

Die Analogeingangsklemmen erscheinen mit insgesamt 9 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 6 Datenbytes sowie drei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 6 Worte im Prozessabbild belegt.

Tabelle 332: 3-Phasen-Leistungsmessklemme

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S0	Statusbyte 0
1	D1	D0	Eingangsdatenwort 1
2	-	S1	Statusbyte 1
3	D3	D2	Eingangsdatenwort 2
4	-	S2	Statusbyte 2
5	D5	D4	Eingangsdatenwort 3

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0	Steuerbyte 0
1	D1	D0	Ausgangsdatenwort 1
2	-	C1	Steuerbyte 1
3	D3	D2	Ausgangsdatenwort 2
4	-	C2	Steuerbyte 2
5	D5	D4	Ausgangsdatenwort 3

13.2.3.5 8-Kanal-Analogeingangsklemmen

750-451

Tabelle 333: 8-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Messwert Kanal 1
1	D3	D2	Messwert Kanal 2
2	D5	D4	Messwert Kanal 3
3	D7	D6	Messwert Kanal 4
4	D9	D8	Messwert Kanal 5
5	D11	D10	Messwert Kanal 6
6	D13	D12	Messwert Kanal 7
7	D15	D14	Messwert Kanal 8

13.2.4 Analogausgangsklemmen

Die Analogausgangsklemmen liefern je Kanal 16-Bit-Ausgabewerte und 8 Steuer-/Statusbits.
MODBUS/TCP verwendet die 8 Steuer-/Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Ausgangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/-controller mit MODBUS/TCP deshalb nur die 16-Bit-Ausgabewerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitalausgangsklemmen gesteckt sind, werden die analogen Ausgangsdaten immer vor die digitalen Daten in das Ausgangs-prozessabbild abgebildet.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Homepage unter: www.wago.com.

13.2.4.1 2-Kanal-Analogausgangsklemmen

750-550, -552, -554, -556, -560, -562, 563, -585, (und alle Varianten),
753-550, -552, -554, -556

Tabelle 334: 2-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2

13.2.4.2 4-Kanal-Analogausgangsklemmen

750-553, -555, -557, -559,
753-553, -555, -557, -559

Tabelle 335: 4-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2
2	D5	D4	Ausgabewert Kanal 3
3	D7	D6	Ausgabewert Kanal 4

13.2.4.3 8-Kanal-Analogausgangsklemmen

Tabelle 336: 8-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Ausgabewert Kanal 1
1	D3	D2	Ausgabewert Kanal 2
2	D5	D4	Ausgabewert Kanal 3
3	D7	D6	Ausgabewert Kanal 4
4	D9	D8	Ausgabewert Kanal 5
5	D11	D10	Ausgabewert Kanal 6
6	D13	D12	Ausgabewert Kanal 7
7	D15	D14	Ausgabewert Kanal 8

13.2.5 Sonderklemmen

Bei einzelnen Klemmen wird neben den Datenbytes auch das Control-/Statusbyte eingeblendet. Dieses dient dem bidirektionalen Datenaustausch der Busklemme mit der übergeordneten Steuerung.

Das Controlbyte wird von der Steuerung an die Klemme und das Statusbyte von der Klemme an die Steuerung übertragen. Somit ist beispielsweise das Setzen eines Zählers mit dem Steuerbyte oder die Anzeige von Bereichsunter- oder -überschreitung durch das Statusbyte möglich.

Das Control-/Statusbyte liegt im Prozessabbild stets im Low-Byte.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der Internetseite www.wago.com.

13.2.5.1 Zählerklemmen

750-404, (und alle Varianten außer /000-005),
753-404, (und Variante /000-003)

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Die Busklemmen liefern dann 32-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 337: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003)

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S	Statusbyte
1	D1	D0	Zählerwert
2	D3	D2	

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C	Steuerbyte
1	D1	D0	Zählersetzwert
2	D3	D2	

750-404/000-005

Die Zählerklemmen belegen insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Diese Busklemmen liefern pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 338: Zählerklemmen 750-404/000-005

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S	Statusbyte
1	D1	D0	Zählerwert Zähler 1
2	D3	D2	Zählerwert Zähler 2

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C	Steuerbyte
1	D1	D0	Zählersetzwert Zähler 1
2	D3	D2	Zählersetzwert Zähler 2

750-638,
753-638

Diese Zählerklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Die Busklemmen liefern dann pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 339: Zählerklemmen 750-638, 753-638

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	S0	Statusbyte von Zähler 1
1	D1	D0	Zählerwert von Zähler 1
2	-	S1	Statusbyte von Zähler 2
3	D3	D2	Zählerwert von Zähler 2

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0	Steuerbyte von Zähler 1
1	D1	D0	Zählersetzwert von Zähler 1
2	-	C1	Steuerbyte von Zähler 2
3	D3	D2	Zählersetzwert von Zähler 2

13.2.5.2 Pulsweitenklemmen

750-511, (und alle Varianten /xxx-xxx)

Diese Pulsweitenklemmen belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 340: Pulsweitenklemmen 750-511, /xxx-xxx

Ein- und Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwert von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwert von Kanal 2

13.2.5.3 Serielle Schnittstellen mit alternativem Datenformat

750-650, (und die Varianten /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (und die Varianten /000-001, -002, -003),
750-653, (und die Varianten /000-002, -007),

753-650, -653

Hinweis

Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebsart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die seriellen Schnittstellenklemmen, die auf das alternative Datenformat eingestellt sind, belegen insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 341: Serielle Schnittstellen mit alternativem Datenformat

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C/S	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	

13.2.5.4 Serielle Schnittstellen mit Standard-Datenformat

750-650/000-001, -014, -015, -016
750-653/000-001, -006

Die seriellen Schnittstellenklemmen, die auf das Standard-Datenformat eingestellt sind, belegen insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 342: Serielle Schnittstellen mit Standard-Datenformat

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C/S	Datenbyte	Steuer-/ Statusbyte
1	D2	D1	Datenbytes	
2	D4	D3		

13.2.5.5 Datenaustauschklemmen

750-654, (und die Variante /000-001)

Die Datenaustauschklemmen belegen jeweils insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 343: Datenaustauschklemmen

Ein- und Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Datenbytes
1	D3	D2	

13.2.5.6 SSI-Geber-Interface-Busklemmen

750-630, (und alle Varianten)

Hinweis

Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebsart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Der Aufbau des Prozessabbilds dieser Busklemme hängt dann davon ab, welche Betriebsart eingestellt ist.

Die SSI-Geber Interface Busklemmen mit Status belegen insgesamt 4 Datenbytes im Eingangsbereich des Prozessabbilds. Dabei werden mit word-alignment insgesamt 2 Worte im Prozessabbild belegt.

Tabelle 344: SSI-Geber Interface Busklemmen mit alternativem Datenformat

Eingangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	D1	D0	Datenbytes
1	D3	D2	

13.2.5.7 Weg- und Winkelmessung

750-631/000-004, -010, -011

Die Busklemme 750-631 belegt 5 Bytes im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 345: Weg- und Winkelmessung 750-631/000-004, --010, -011

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S	nicht genutzt	Statusbyte
1	D1	D0	Zählerwort	
2	-	-	nicht genutzt	
3	D4	D3	Latchwort	

Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C	Steuerbyte von Zähler 1
1	D1	D0	Zählersetzwert von Zähler 1
2	-	-	nicht genutzt
3	-	-	nicht genutzt

750-634

Die Busklemme 750-634 belegt 5 Bytes (in der Betriebsart Periodendauermessung mit 6 Bytes) im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 346: Incremental-Encoder-Interface 750-634

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S	nicht genutzt	Statusbyte
1	D1	D0	Zählerwort	
2	-	(D2) *)	nicht genutzt	(Periodendauer)
3	D4	D3	Latchwort	

*) Ist durch das Steuerbyte die Betriebsart Periodendauermessung eingestellt, wird in D2 zusammen mit D3/D4 die Periodendauer als 24-Bit-Wert ausgegeben.

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C	nicht genutzt	Steuerbyte
1	D1	D0	Zählersetzwort	
2	-	-	nicht genutzt	
3	-	-		

750-637

Die Incremental-Encoder-Interface Busklemme belegt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 347: Incremental-Encoder-Interface 750-637

Ein- und Ausgangsprozessabbild			
Offset	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
0	-	C0/S0	Steuer-/Statusbyte von Kanal 1
1	D1	D0	Datenwerte von Kanal 1
2	-	C1/S1	Steuer-/Statusbyte von Kanal 2
3	D3	D2	Datenwerte von Kanal 2

750-635,
753-635

Die Digitale Impuls Schnittstelle belegt insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 348: Digitale Impuls Schnittstelle 750-635

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C0/S0	Datenbyte	Steuer-/Statusbyte
1	D2	D1	Datenbytes	

13.2.5.8 DC-Drive Controller

750-636

Der DC-Drive-Controller 750-636 stellt dem Koppler über 1 logischen Kanal 6 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Positionsdaten werden in 4 Ausgangsbytes (D0 ... D3) und 4 Eingangsbytes (D0 ... D3) abgelegt. 2 Steuerbytes (C0, C1) und 2 Statusbytes (S0, S1) dienen zur Steuerung der Busklemme und des Antriebs. Alternativ zu den Positionsdaten im Eingangsprozessabbild (D0 ... D3) können erweiterte Statusinformationen (S2 ... S5) eingeblendet werden. Die 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen den Prozessdaten und den erweiterten Statusbytes im Eingangsprozessabbild erfolgt über Bit 3 (ExtendedInfo_ON) im Controlbyte C1 (C1.3). Mit Bit 3 des Statusbytes S1 (S1.3) wird die Umschaltung quittiert.

Tabelle 349: Antriebssteuerung 750-636

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	S1	S0	Status S1	Statusbyte S0
1	D1*) / S3**)	D0*) / S2**)	Istposition*) / Erweitertes Statusbyte S3**)	Istposition (LSB)*) / Erweitertes Statusbyte S2**)
2	D3*) / S5**)	D2*) / S4**)	Istposition (MSB)*) / Erweitertes Statusbyte S3**)	Istposition*) / Erweitertes Statusbyte S4**)

*) ExtendedInfo_ON = ‘0’.
**) ExtendedInfo_ON = ‘1’.

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	C1	C0	Steuerbyte C1	Steuerbyte C0
1	D1	D0	Sollposition	Sollposition (LSB)
2	D3	D2	Sollposition (MSB)	Sollposition

13.2.5.9 Steppercontroller

750-670

Der Steppercontroller RS 422 / 24 V / 20 mA 750-670 stellt dem Feldbuskoppler über 1 logischen Kanal 12 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Daten werden in Abhängigkeit von der Betriebsart in bis zu 7 Ausgangsbytes (D0 ... D6) und 7 Eingangsbytes (D0 ... D6) abgelegt. Das Ausgangsbyte D0 und das Eingangsbyte D0 sind reserviert und ohne Funktion. Ein Klemmenbus-Steuer- und Statusbyte (C0, S0) sowie 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen beiden Prozessabbildern erfolgt über das Bit 5 im Controlbyte C0 (C0.5). Mit dem Bit 5 des Statusbytes S0 (S0.5) wird das Einschalten der Mailbox quittiert.

Tabelle 350: Steppercontroller RS 422 / 24 V / 20 mA 750-670

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	Reserviert	S0	Reserviert	Statusbyte S0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	S3	D6	Statusbyte S3	Prozessdaten*) / Reserviert**)
5	S1	S2	Statusbyte S1	Statusbyte S2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	Reserviert	C0	Reserviert	Controlbyte C0
1	D1	D0	Prozessdaten*) / Mailbox**)	
2	D3	D2		
3	D5	D4		
4	C3	D6	Controlbyte C3	Prozessdaten*) / Reserviert**)
5	C1	C2	Controlbyte C1	Controlbyte C2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

13.2.5.10 RTC-Modul

750-640

Das RTC-Modul belegt insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes, ein zusätzliches Steuer-/ Statusbyte und jeweils ein Befehlsbyte (ID). Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 351: RTC-Modul 750-640

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	ID	C/S	Befehlsbyte	Steuer-/ Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		

13.2.5.11 DALI/DSI-Masterklemme

750-641

Die DALI/DSI-Masterklemme belegt insgesamt 6 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 352: DALI/DSI-Masterklemme 750-641

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	S	DALI-Antwort	Statusbyte
1	D2	D1	Message 3	DALI-Adresse
2	D4	D3	Message 1	Message 2

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	C	DALI-Befehl, DSI-Dimmwert	Steuerbyte
1	D2	D1	Parameter 2	DALI-Adresse
2	D4	D3	Command-Extension	Parameter 1

13.2.5.12 DALI-Multi-Master-Klemme

753-647

Die DALI-Multi-Master-Klemme belegt insgesamt 24 Byte im Ein- und Ausgangsbereich des Prozessabbildes.

Die DALI-Multi-Master-Klemme kann im „Easy-Modus“ (Standardeinstellung) und im „Full-Modus“ betrieben werden. Der „Easy-Modus“ wird zur Übermittlung einfacher binärer Signale für die Beleuchtungssteuerung verwendet. Eine Konfiguration oder Programmierung mittels DALI-Masterbaustein ist im „Easy-Modus“ nicht notwendig.

Veränderungen von einzelnen Bits des Prozessabbildes werden direkt in DALI-Kommandos für ein vorkonfiguriertes DALI-Netzwerk umgewandelt. Von dem 24-Byte-Prozessabbild können im „Easy-Modus“ 22 Bytes direkt zum Schalten von EVGs, Gruppen oder Szenen genutzt werden. Schaltbefehle werden über DALI- und Gruppenadressen übertragen, dabei wird jede DALI- und jede Gruppenadresse durch ein 2-Bit-Paar repräsentiert.

Der Aufbau der Prozessdaten ist im Einzelnen in den anschließenden Tabellen dargestellt.

Tabelle 353: Übersicht über das Eingangsprozessabbild im „Easy-Modus“

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S	res.	Status Broadcast schalten: Bit 0: 1-/2-Tasten-Modus Bit 2: Broadcast-Status EIN/AUS Bit 1,3-7: -
1	DA4…DA7	DA0…DA3	Bitpaar für DALI-Adresse DA0:	
2	DA12…DA15	DA8…DA11	Bit 1: Bit gesetzt = EIN	
3	DA20…DA23	DA16…DA19	Bit nicht gesetzt = AUS	
4	DA28…DA31	DA24…DA27	Bit 2: Bit gesetzt = Fehler	
5	DA36…DA39	DA32…DA35	Bit nicht gesetzt = kein Fehler	
6	DA44…DA47	DA40…DA43	Bitpaare DA1 bis DA63 analog zu DA0.	
7	DA52…DA55	DA48…DA51		
8	DA60…DA63	DA56…DA59		
9	GA4…GA7	GA0…GA3	Bitpaar für DALI-Gruppenadresse GA0: Bit 1: Bit gesetzt = EIN Bit nicht gesetzt = AUS	
10	GA12…GA15	GA8…GA11	Bit 2: Bit gesetzt = Fehler Bit nicht gesetzt = kein Fehler Bitpaare GA1 bis GA15 analog zu GA0.	
11	-	-	nicht verwendet	

DA = DALI-Adresse
GA = Gruppenadresse

Tabelle 354: Übersicht über das Ausgangsprozessabbild im „Easy-Modus“

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S	res.	Broadcast EIN/AUS und schalten: Bit 0: Broadcast EIN Bit 1: Broadcast AUS Bit 2: Broadcast EIN/AUS/dimmen Bit 3: Broadcast kurz EIN/AUS Bit 4…7: reserviert
1	DA4…DA7	DA0…DA3	Bitpaar für DALI-Adresse DA0:	
2	DA12…DA15	DA8…DA11	Bit 1: kurz: DA schalten EIN	
3	DA20…DA23	DA16…DA19	lang: dimmen, heller	
4	DA28…DA31	DA24…DA27	Bit 2: kurz: DA schalten AUS	
5	DA36…DA39	DA32…DA35	lang: dimmen, dunkler	
6	DA44…DA47	DA40…DA43	Bitpaare DA1 bis DA63 analog zu DA0.	
7	DA52…DA55	DA48…DA51		
8	DA60…DA63	DA56…DA59		
9	GA4…GA7	GA0…GA3	Bitpaar für DALI-Gruppenadresse GA0: Bit 1: kurz: GA schalten EIN lang: dimmen heller	
10	GA12…GA15	GA8…GA11	Bit 2: kurz: GA schalten AUS lang: dimmen dunkler Bitpaare GA1 bis GA15 analog zu GA0.	
11	Bit 8…15	Bit 0…7	Szene 0…15 schalten	

DA = DALI-Adresse
GA = Gruppenadresse

13.2.5.13 LON®-FTT-Klemme

753-648

Das Prozessabbild der LON®-FTT-Klemme besteht aus einem Steuer-/Statusbyte und 23 Byte bidirektionaler Kommunikationsdaten, die von dem WAGO-I/O-*PRO*- Funktionsbaustein „LON_01.lib“ verarbeitet werden. Dieser Baustein ist für die Funktion der LON®-FTT-Klemme unbedingt erforderlich und stellt steuerungsseitig eine Anwenderschnittstelle zur Verfügung.

13.2.5.14 Funkreceiver EnOcean

750-642

Die EnOcean Funkreceiverklemme belegt insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Die 3 Bytes Ausgangsdaten werden jedoch nicht genutzt. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 355: Funkreceiver EnOcean 750-642

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	D0	S	Datenbyte	Statusbyte
1	D2	D1	Datenbytes	

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C	nicht genutzt	Steuerbyte
1	-	-	nicht genutzt	

13.2.5.15 MP-Bus-Masterklemme

750-643

Die MP-Bus-Masterklemme belegt insgesamt 8 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 6 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 356: MP-Bus-Masterklemme 750-643

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	C1/S1	C0/S0	erweitertes Steuer-/Statusbyte	Steuer-/Statusbyte
1	D1	D0	Datenbytes	
2	D3	D2		
3	D5	D4		

13.2.5.16 *Bluetooth*® RF-Transceiver

750-644

Die Größe des Prozessabbildes der *Bluetooth*®-Busklemme ist in den festgelegten Größen 12, 24 oder 48 Byte einstellbar.
Es besteht aus einem Steuerbyte (Eingang) bzw. Statusbyte (Ausgang), einem Leerbyte, einer 6, 12 oder 18 Byte großen, überlagerbaren Mailbox (Modus 2) und den *Bluetooth*®-Prozessdaten in einem Umfang von 4 bis 46 Byte.

Die *Bluetooth*®-Busklemme belegt also jeweils 12 bis maximal 48 Bytes im Prozessabbild, wobei die Größen des Eingangs- und Ausgangsprozessabbildes stets übereinstimmen.

Das erste Byte enthält das Steuer-/Statusbyte, das zweite ein Leerbyte.
Daran schließen sich bei ausgeblendeter Mailbox unmittelbar Prozessdaten an.
Bei eingeblendeter Mailbox werden je nach deren Größe die ersten 6, 12 oder 18 Byte Prozessdaten von Mailbox-Daten überlagert. Die Bytes im Bereich hinter der optional einblendbaren Mailbox enthalten grundsätzlich Prozessdaten. Den internen Aufbau der *Bluetooth*®-Prozessdaten entnehmen Sie der Dokumentation des *Bluetooth*® RF-Transceivers 750-644.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 357: Bluetooth® RF-Transceiver 750-644

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
1	D1	D0	Mailbox (0, 3, 6 oder 9 Worte) sowie Prozessdaten (2-23 Worte)	
2	D3	D2		
3	D5	D4		
...	...	...		
max. 23	D45	D44		

13.2.5.17 Schwingstärke/Wälzlagerüberwachung VIB I/O

750-645

Die Schwingstärke/Wälzlagerüberwachung VIB I/O belegt insgesamt 12 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 8 Datenbytes und vier zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 8 Worte im Prozessabbild belegt.

Tabelle 358: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte (log. Kanal 1, Sensoreingang 1)
1	D1	D0	Datenbytes (log. Kanal 1, Sensoreingang 1)	
2	-	C1/S1	nicht genutzt	Steuer-/Statusbyte (log. Kanal 2, Sensoreingang 2)
3	D3	D2	Datenbytes (log. Kanal 2, Sensoreingang 2)	
4	-	C2/S2	nicht genutzt	Steuer-/Statusbyte (log. Kanal 3, Sensoreingang 3)
5	D5	D4	Datenbytes (log. Kanal 3, Sensoreingang 3)	
6	-	C3/S3	nicht genutzt	Steuer-/Statusbyte (log. Kanal 4, Sensoreingang 4)
7	D7	D6	Datenbytes (log. Kanal 4, Sensoreingang 4)	

13.2.5.18 KNX/EIB/TP1-Klemme

753-646

Die KNX/TP1-Klemme erscheint im Router- sowie im Gerätemodus mit insgesamt 24 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbildes, 20 Datenbytes und 1 Steuer-/Statusbyte. Die zusätzlichen Bytes S1 bzw. C1 werden als Datenbytes transferiert, aber als erweiterte Status- und Steuerbytes verwendet. Der Opcode dient als Schreib- und Lesekommando für Daten oder als Auslöser bestimmter Funktionen der KNX/EIB/TP1-Klemme. Mit word-alignment werden jeweils 12 Worte im Prozessabbild belegt. Im Routermodus ist kein Zugriff auf das Prozessabbild möglich. Telegramme werden nur getunnelt übertragen.

Im Gerätemodus erfolgt der Zugriff auf KNX-Daten über spezielle Funktionsbausteine der IEC-Applikation. Eine Konfiguration mittels der allgemeinen Engineering-Tool-Software (ETS) für KNX ist notwendig.

Tabelle 359: KNX/EIB/TP1-Klemme 753-646

Eingangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	S0	nicht genutzt	Statusbyte
1	S1	OP	Erweitertes Statusbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0	nicht genutzt	Steuerbyte
1	C1	OP	Erweitertes Steuerbyte	Opcode
2	D1	D0	Datenbyte 1	Datenbyte 0
3	D3	D2	Datenbyte 3	Datenbyte 2
4	D5	D4	Datenbyte 5	Datenbyte 4
5	D7	D6	Datenbyte 7	Datenbyte 6
6	D9	D8	Datenbyte 9	Datenbyte 8
7	D11	D10	Datenbyte 11	Datenbyte 10
8	D13	D12	Datenbyte 13	Datenbyte 12
9	D15	D14	Datenbyte 15	Datenbyte 14
10	D17	D16	Datenbyte 17	Datenbyte 16
11	D19	D18	Datenbyte 19	Datenbyte 18

13.2.5.19 AS-Interface-Masterklemme

750-655

Das Prozessabbild der AS-Interface-Masterklemme ist in seiner Länge einstellbar in den festgelegten Größen von 12, 20, 24, 32, 40 oder 48 Byte.
Es besteht aus einem Control- bzw. Statusbyte, einer 0, 6, 10, 12 oder 18 Byte großen Mailbox und den AS-interface Prozessdaten in einem Umfang von 0 bis 32 Byte.

Mit word-alignment belegt die AS-Interface-Masterklemme also jeweils 6 bis maximal 24 Worte im Prozessabbild.

Das erste Ein- bzw. Ausgangswort enthält das Status- bzw. Controlbyte sowie ein Leerbyte.
Daran schließen sich für die fest eingeblendete Mailbox (Modus 1) die Worte mit Mailboxdaten an.

Wenn die Mailbox überlagerbar eingestellt ist (Modus 2), enthalten diese Worte Mailbox- oder Prozessdaten.
Die weiteren Worte enthalten die restlichen Prozessdaten.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 360: AS-Interface-Masterklemme 750-655

Ein- und Ausgangsprozessabbild				
Offset	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
0	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
1	D1	D0	Mailbox (0, 3, 5, 6 oder 9 Worte) sowie Prozessdaten (0-16 Worte)	
2	D3	D2		
3	D5	D4		
...	...	...		
max. 23	D45	D44		

13.2.6 Systemklemmen

13.2.6.1 Systemklemmen mit Diagnose

750-610, -611

Die Potentialeinspeiseklemmen 750-610 und -611 mit Diagnose liefern zur Überwachung der Versorgung 2 Bits Diagnosedaten.

Tabelle 361: Systemklemmen mit Diagnose 750-610, -611

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 2 Sicherung	Diagnosebit S 1 Spannung

13.2.6.2 Binäre Platzhalterklemmen

750-622

Die binären Platzhalterklemmen 750-622 verhalten sich wahlweise wie 2-Kanal-Digitaleingangs- oder -ausgangsklemmen und belegen je nach angewählter Einstellung pro Kanal 1, 2, 3 oder 4 Bits.
Dabei werden dann entsprechend 2, 4, 6 oder 8 Bits entweder im Prozesseingangs- oder -ausgangsabbild belegt.

Tabelle 362: Binäre Platzhalterklemmen 750-622 (mit dem Verhalten einer 2 DI)

Ein- oder Ausgangsgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
(Datenbit DI 8)	(Datenbit DI 7)	(Datenbit DI 6)	(Datenbit DI 5)	(Datenbit DI 4)	(Datenbit DI 3)	Datenbit DI 2	Datenbit DI 1

13.3 Aufbau der Prozessdaten für EtherNet/IP

Der Aufbau der Prozessdaten ist bei einigen Busklemmen bzw. deren Varianten feldbusspezifisch.

Bei dem Feldbuscontroller mit EtherNet/IP wird das Prozessabbild wortweise aufgebaut (mit word-alignment). Die interne Darstellung der Daten, die größer als ein Byte sind, erfolgt nach dem Intel-Format.

Im Folgenden wird für alle Busklemmen des WAGO-I/O-SYSTEMs 750 und 753 die feldbusspezifische Darstellung im Prozessabbild des Feldbuscontrollers mit EtherNet/IP beschrieben und der Aufbau der Prozesswerte gezeigt.

Für das PFC-Prozessabbild des Feldbuscontrollers ist der Aufbau der Prozesswerte identisch.

ACHTUNG

Geräteschäden durch falsche Adressierung!
Zur Vermeidung von Geräteschäden im Feldbereich, müssen Sie bei der Adressierung einer an beliebiger Position im Feldbusknoten befindlichen Busklemme, die Prozessdaten aller vorherigen byte- bzw. bitweise-orientierten Busklemmen berücksichtigen.

13.3.1 Digitaleingangsklemmen

Die Digitaleingangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Signalzustand des jeweiligen Kanals angibt. Diese Bits werden in das Eingangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie z. B. Drahtbruch und/oder Kurzschluss.

Sofern in dem Knoten auch Analogeingangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Eingangsdaten in dem Eingangsprozessabbild angehängt.
Dabei wird für jeweils 8 Bit ein Subindex belegt.

Jeder Eingangskanal belegt in dem Discrete Input Point Object (Class 0x65) eine Instanz.

13.3.1.1 1-Kanal-Digitaleingangsklemmen mit Diagnose

750-435

Tabelle 363: 1-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 1	Datenbit DI 1

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.

13.3.1.2 2-Kanal-Digitaleingangsklemmen

750-400, -401, -405, -406, -410, -411, -412, -427, -438, (und alle Varianten),
753-400, -401, -405, -406, -410, -411, -412, -427

Tabelle 364: 2-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.

13.3.1.3 2-Kanal-Digitaleingangsklemmen mit Diagnose

750-419, -421, -424, -425
753-421, -424, -425

Tabelle 365: 2-Kanal-Digitaleingangsklemmen mit Diagnose

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

13.3.1.4 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

750-418
753-418

Die Digitaleingangsklemme 750-418, 753-418 liefert über die Prozesswerte im Eingangsprozessabbild hinaus 4 Bit Daten, die im Ausgangsprozessabbild dargestellt werden.

Tabelle 366: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Quittierungsbit Q 2 Kanal 2	Quittierungsbit Q 1 Kanal 1	0	0

Die Eingangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.1.5 4-Kanal-Digitaleingangsklemmen

750-402, -403, -408, -409, -414, -415, -422, -423, -428, -432, -433, -1420, -1421, -1422
753-402, -403, -408, -409, -415, -422, -423, -428, -432, -433, -440

Tabelle 367: 4-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
				Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 4 Instanzen.

13.3.1.6 8-Kanal-Digitaleingangsklemmen

750-430, -431, -436, -437, -1415, -1416, -1417
753-430, -431, -434

Tabelle 368: 8-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Eingangsklemmen belegen in Klasse (0x65) 8 Instanzen.

13.3.1.7 16-Kanal-Digitaleingangsklemmen

750-1400, -1402, -1405, -1406, -1407

Tabelle 369: 16-Kanal-Digitaleingangsklemmen

Eingangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 16 Kanal 16	Datenbit DI 15 Kanal 15	Datenbit DI 14 Kanal 14	Datenbit DI 13 Kanal 13	Datenbit DI 12 Kanal 12	Datenbit DI 11 Kanal 11	Datenbit DI 10 Kanal 10	Datenbit DI 9 Kanal 9	Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

13.3.2 Digitalausgangsklemmen

Die Digitalausgangsklemmen liefern als Prozesswerte pro Kanal je ein Bit, das den Status des jeweiligen Kanals angibt. Diese Bits werden in das Ausgangsprozessabbild gemappt.

Einzelne digitale Busklemmen stellen sich mit einem zusätzlichen Diagnosebit pro Kanal im Eingangsprozessabbild dar. Das Diagnosebit dient zur Auswertung eines auftretenden Fehlers, wie Drahtbruch und/oder Kurzschluss. Bei einigen Busklemmen müssen, bei gesetztem Diagnosebit, zusätzlich die Datenbits ausgewertet werden.

Sofern in dem Knoten auch Analogausgangsklemmen gesteckt sind, werden die digitalen Daten immer, byteweise zusammengefasst, hinter die analogen Ausgangsdaten in dem Ausgangsprozessabbild angehängt.
Dabei wird für jeweils 8 Bit ein Subindex belegt.

Jeder Ausgangskanal belegt in dem Discrete Output Point Object (Class 0x66) eine Instanz.

13.3.2.1 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

750-523

Die Digitalausgangsklemmen liefern über das Prozesswerte Bit im Ausgangsprozessabbild hinaus 1 Bit, das im Eingangsprozessabbild dargestellt wird. Dieses Statusbit zeigt den „Handbetrieb“ an.

Tabelle 370: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	Statusbit „Hand-betrieb“

Die Ausgangsklemmen belegen in Klasse (0x65) 2 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						nicht genutzt	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

13.3.2.2 2-Kanal-Digitalausgangsklemmen

750-501, -502, -509, -512, -513, -514, -517, -535, (und alle Varianten),
753-501, -502, -509, -512, -513, -514, -517

Tabelle 371: 2-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

13.3.2.3 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-507 (-508), -522,
753-507

Die Digitalausgangsklemmen liefern über die 2-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 2 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 372: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x65) 2 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
						steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 2 Instanzen.

750-506,
753-506

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die durch einen 2-Bit Fehlercode eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 373: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 3 Kanal 2	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1	Diagnosebit S 0 Kanal 1
Diagnosebits S1/S0, S3/S2: = ‘00’ normaler Betrieb Diagnosebits S1/S0, S3/S2: = ‘01’ keine Last angeschlossen/Kurzschluss gegen +24 V Diagnosebits S1/S0, S3/S2: = ‘10’ Kurzschluss gegen GND/Überlast							

Die Ausgangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				nicht genutzt	nicht genutzt	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.4 4-Kanal-Digitalausgangsklemmen

750-504, -516, -519, -531,
753-504, -516, -531, -540

Tabelle 374: 4-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.5 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-532

Die Digitalausgangsklemmen liefern über die 4-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 4 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 375: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1
Diagnosebit S = ‘0’ kein Fehler Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast							

Die Ausgangsklemmen belegen in Klasse (0x65) 4 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
				steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 4 Instanzen.

13.3.2.6 8-Kanal-Digitalausgangsklemmen

750-530, -536, -1515, -1516
753-530, -534

Tabelle 376: 8-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.

13.3.2.7 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

750-537

Die Digitalausgangsklemmen liefern über die 8-Bit-Prozesswerte im Ausgangsprozessabbild hinaus 8 Bit Daten, die im Eingangsprozessabbild dargestellt werden. Dieses sind kanalweise zugeordnete Diagnosebits, die eine Überlast, einen Kurzschluss oder einen Drahtbruch anzeigen.

Tabelle 377: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Diagnosebit S 8 Kanal 8	Diagnosebit S 7 Kanal 7	Diagnosebit S 6 Kanal 6	Diagnosebit S 5 Kanal 5	Diagnosebit S 4 Kanal 4	Diagnosebit S 3 Kanal 3	Diagnosebit S 2 Kanal 2	Diagnosebit S 1 Kanal 1
Diagnosebit S = ‘0’ kein Fehler Diagnosebit S = ‘1’ Drahtbruch, Kurzschluss oder Überlast							

Die Ausgangsklemmen belegen in Klasse (0x65) 8 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.

13.3.2.8 16-Kanal-Digitalausgangsklemmen

750-1500, -1501, -1504, -1505

Tabelle 378: 16-Kanal-Digitalausgangsklemmen

Ausgangsprozessabbild															
Bit 15	Bit 14	Bit 13	Bit 12	Bit 11	Bit 10	Bit 9	Bit 8	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 16 Kanal 16	steuert DO 15 Kanal 15	steuert DO 14 Kanal 14	steuert DO 13 Kanal 13	steuert DO 12 Kanal 12	steuert DO 11 Kanal 11	steuert DO 10 Kanal 10	steuert DO 9 Kanal 9	steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ausgangsklemmen belegen in Klasse (0x66) 16 Instanzen.

13.3.2.9 8-Kanal-Digitalein- / -ausgangsklemmen

750-1502, -1506

Tabelle 379: 8-Kanal-Digitalein-/ -ausgangsklemmen

Eingangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Datenbit DI 8 Kanal 8	Datenbit DI 7 Kanal 7	Datenbit DI 6 Kanal 6	Datenbit DI 5 Kanal 5	Datenbit DI 4 Kanal 4	Datenbit DI 3 Kanal 3	Datenbit DI 2 Kanal 2	Datenbit DI 1 Kanal 1

Die Ein-/Ausgangsklemmen belegen in Klasse (0x65) 8 Instanzen.

Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
steuert DO 8 Kanal 8	steuert DO 7 Kanal 7	steuert DO 6 Kanal 6	steuert DO 5 Kanal 5	steuert DO 4 Kanal 4	steuert DO 3 Kanal 3	steuert DO 2 Kanal 2	steuert DO 1 Kanal 1

Die Ein-/Ausgangsklemmen belegen in Klasse (0x66) 8 Instanzen.

13.3.3 Analogeingangsklemmen

Die Analogeingangsklemmen liefern je Kanal 16-Bit-Messwerte und 8 Steuer-/ Statusbits.

EtherNet/IP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Eingangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/ -controller mit EtherNet/IP deshalb nur die 16-Bit-Messwerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitaleingangsklemmen gesteckt sind, werden die analogen Eingangsdaten immer vor die digitalen Daten in das Eingangs-prozessabbild abgebildet.

Jeder Eingangskanal belegt in dem Analog Input Point Object (Class 0x67) eine Instanz.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

13.3.3.1 1-Kanal-Analogeingangsklemmen

750-491, (und alle Varianten)

Tabelle 380: 1-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	Bezeichnung der Bytes		Bemerkung
	High Byte	Low Byte	
n	D1	D0	Messwert U_D
n+1	D3	D2	Messwert U_{ref}

Diese Eingangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

13.3.3.2 2-Kanal-Analogeingangsklemmen

750-452, -454, -456, -461, -462, -465, -466, -467, -469, -472, -474, -475, 476, -477, -478, -479, -480, -481, -483, -485, -492, (und alle Varianten),
753-452, -454, -456, -461, -465, -466, -467, -469, -472, -474, -475, 476, -477, 478, -479, -483, -492, (und alle Varianten)

Tabelle 381: 2-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2

Diese Eingangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

13.3.3.3 4-Kanal-Analogeingangsklemmen

750-453, -455, -457, -459, -460, -468, (und alle Varianten),
753-453, -455, -457, -459

Tabelle 382: 4-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2
n+2	D5	D4	Messwert Kanal 3
n+3	D7	D6	Messwert Kanal 4

Diese Eingangsklemmen stellen sich mit 4x2 Byte dar und belegen in Klasse (0x67) 4 Instanzen.

13.3.3.4 4-Kanal-Analogeingangsklemmen für Widerstandssensoren (RTD)

750-450

Tabelle 383: 4-Kanal-Analogeingangsklemmen für Widerstandssensoren [RTD]

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2
n+2	D5	D4	Messwert Kanal 3
n+3	D7	D6	Messwert Kanal 4

Diese Eingangsklemmen stellen sich mit 4x2 Byte dar und belegen in Klasse (0x67) 4 Instanzen.

13.3.3.5 3-Phasen-Leistungsmessklemme

750-493

Die Analogeingangsklemmen erscheinen mit insgesamt 9 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 6 Datenbytes sowie drei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 6 Worte im Prozessabbild belegt.

Tabelle 384: 3-Phasen-Leistungsmessklemme

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S0	Statusbyte 0
	D1	D0	Eingangsdatenwort 1
n+1	-	S1	Statusbyte 1
	D3	D2	Eingangsdatenwort 2
n+2	-	S2	Statusbyte 2
	D5	D4	Eingangsdatenwort 3

Diese Analogeingangsklemmen stellen sich mit 3x4 Bytes dar und belegen in Klasse (0x67) 3 Instanzen.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0	Steuerbyte 0
	D1	D0	Ausgangsdatenwort 1
n+1	-	C1	Steuerbyte 1
	D3	D2	Ausgangsdatenwort 2
n+2	-	C2	Steuerbyte 2
	D5	D4	Ausgangsdatenwort 3

Diese Analogeingangsklemmen stellen sich mit 3x4 Bytes dar und belegen in Klasse (0x68) 3 Instanzen.

13.3.3.6 8-Kanal-Analogeingangsklemmen

750-451

Tabelle 385: 8-Kanal-Analogeingangsklemmen

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Messwert Kanal 1
n+1	D3	D2	Messwert Kanal 2
n+2	D5	D4	Messwert Kanal 3
n+3	D7	D6	Messwert Kanal 4
n+4	D9	D8	Messwert Kanal 5
n+5	D11	D10	Messwert Kanal 6
n+6	D13	D12	Messwert Kanal 7
n+7	D15	D14	Messwert Kanal 8

Diese Eingangsklemmen stellen sich mit 8x2 Bytes dar und belegen in Klasse (0x67) 8 Instanzen.

13.3.4 Analogausgangsklemmen

Die Analogausgangsklemmen liefern je Kanal 16-Bit-Ausgabewerte und 8 Steuer-/ Statusbits.
EtherNet/IP verwendet die 8 Steuer-/ Statusbits jedoch nicht, d. h. es erfolgt kein Zugriff und keine Auswertung.

In das Ausgangsprozessabbild für den Feldbus werden bei dem Feldbuskoppler/-controller mit EtherNet/IP deshalb nur die 16-Bit-Ausgabewerte pro Kanal im Intel-Format und wortweise gemappt.

Sofern in dem Knoten auch Digitalausgangsklemmen gesteckt sind, werden die analogen Ausgangsdaten immer vor die digitalen Daten in das Ausgangs-prozessabbild abgebildet.

Jeder Ausgangskanal belegt in dem Analog Output Point Object (Class 0x68) eine Instanz.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

13.3.4.1 2-Kanal-Analogausgangsklemmen

750-550, -552, -554, -556, -560, -562, 563, -585, (und alle Varianten),
753-550, -552, -554, -556

Tabelle 386: 2-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Ausgabewert Kanal 1
n+1	D3	D2	Ausgabewert Kanal 2

Diese Ausgangsklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x68) 2 Instanzen.

13.3.4.2 4-Kanal-Analogausgangsklemmen

750-553, -555, -557, -559,
753-553, -555, -557, -559

Tabelle 387: 4-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Ausgabewert Kanal 1
n+1	D3	D2	Ausgabewert Kanal 2
n+2	D5	D4	Ausgabewert Kanal 3
n+3	D7	D6	Ausgabewert Kanal 4

Diese Ausgangsklemmen stellen sich mit 4x2 Byte dar und belegen in Klasse (0x68) 4 Instanzen.

13.3.4.3 8-Kanal-Analogausgangsklemmen

Tabelle 388: 8-Kanal-Analogausgangsklemmen

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Ausgabewert Kanal 1
n+1	D3	D2	Ausgabewert Kanal 2
n+2	D5	D4	Ausgabewert Kanal 3
n+3	D7	D6	Ausgabewert Kanal 4
n+4	D9	D8	Ausgabewert Kanal 5
n+5	D11	D10	Ausgabewert Kanal 6
n+6	D13	D12	Ausgabewert Kanal 7
n+7	D15	D14	Ausgabewert Kanal 8

Diese Ausgangsklemmen stellen sich mit 8x2 Byte dar und belegen in Klasse (0x68) 8 Instanzen.

13.3.5 Sonderklemmen

Bei einzelnen Klemmen wird neben den Datenbytes auch das Control-/ Statusbyte eingeblendet. Dieses dient dem bidirektionalen Datenaustausch der Busklemme mit der übergeordneten Steuerung.

Das Control- bzw. Steuerbyte wird von der Steuerung an die Klemme und das Statusbyte von der Klemme an die Steuerung übertragen. Somit ist beispielsweise das Setzen eines Zählers mit dem Steuerbyte oder die Anzeige von Bereichsunter- oder -überschreitung durch das Statusbyte möglich.

Das Control-/Statusbyte liegt bei dem Feldbuskoppler/-controller mit EtherNet/IP stets im Low-Byte.

Information

Informationen zum Steuer-/Statusbyteaufbau
Den speziellen Aufbau der jeweiligen Steuer-/Statusbytes entnehmen Sie bitte der zugehörigen Busklemmenbeschreibung. Ein Handbuch mit der jeweiligen Beschreibung zu jeder Busklemme finden Sie auf der WAGO-Internetseite unter: http://www.wago.com.

Die Sonderklemmen stellen sich wie analoge Klemmen dar.
Deshalb belegen deren Prozesseingangswerte pro Kanal ebenfalls eine Instanz in dem Analog Input Point Object (Class 0x67) und deren Prozessausgangswerte pro Kanal eine Instanz in dem Analog Input Point Object (Class 0x68).

13.3.5.1 Zählerklemmen

750-404, (und alle Varianten außer /000-005),
753-404, (und Variante /000-003)

Die Zählerklemmen erscheinen mit insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Die Busklemmen liefern dann 32-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 389: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003)

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S	Statusbyte
	D1	D0	Zählerwert
	D3	D2	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C	Steuerbyte
	D1	D0	Zählersetzwert
	D3	D2	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

750-404/000-005

Die Zählerklemmen erscheinen mit insgesamt 5 Bytes Nutzdaten im Ein- und Ausgangsbereich der Prozessabbilder, 4 Datenbytes sowie ein zusätzliches Steuer-/Statusbyte. Diese Busklemmen liefern pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 390: Zählerklemmen 750-404/000-005

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S	Statusbyte
	D1	D0	Zählerwert Zähler 1
	D3	D2	Zählerwert Zähler 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C	Steuerbyte
	D1	D0	Zählersetzwert Zähler 1
	D3	D2	Zählersetzwert Zähler 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-638,
753-638

Diese Zählerklemmen erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Die Busklemmen liefern dann pro Zähler 16-Bit-Zählerstände. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 391: Zählerklemmen 750-638, 753-638

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	S0	Statusbyte von Zähler 1
	D1	D0	Zählerwert von Zähler 1
n+1	-	S1	Statusbyte von Zähler 2
	D3	D2	Zählerwert von Zähler 2

Diese Sonderklemmen stellen sich mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0	Steuerbyte von Zähler 1
	D1	D0	Zählersetzwert von Zähler 1
n+1	-	C1	Steuerbyte von Zähler 2
	D3	D2	Zählersetzwert von Zähler 2

Diese Sonderklemmen stellen sich mit 2x3 Byte dar und belegen in Klasse (0x68) 2 Instanzen.

13.3.5.2 Pulsweitenklemmen

750-511, (und alle Varianten /xxx-xxx)

Diese Pulsweitenklemmen erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes sowie zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 392: Pulsweitenklemmen 750-511, /xxx-xxx

Ein- und Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0/S0	Steuer-/Statusbyte von Kanal 1
	D1	D0	Datenwert von Kanal 1
n+1			Steuer-/Statusbyte von Kanal 2
	D3	D2	Datenwert von Kanal 2

Diese Sonderklemmen stellen sich jeweils mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.3 Serielle Schnittstellen mit alternativem Datenformat

750-650, (und die Varianten /000-002, -004, -006, -009, -010, -011, -012, -013),
750-651, (und die Varianten /000-002, -003),
750-653, (und die Varianten /000-002, -007),
753-650, -653

Hinweis

Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Abhängig davon, ist das Prozessabbild dieser Busklemmen dann das gleiche, wie das von der entsprechenden Variante.

Die seriellen Schnittstellenklemmen, die mit dem alternativen Datenformat eingestellt sind, erscheinen mit insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 393: Serielle Schnittstellen mit alternativem Datenformat

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	C/S	Datenbyte	Steuer-/Statusbyte
n+1	D2	D1	Datenbytes	

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.4 Serielle Schnittstellen mit Standard Datenformat

750-650/000-001, -014, -015, -016
750-651/000-001
750-653/000-001, -006

Die seriellen Schnittstellenklemmen, die mit dem Standard Datenformat eingestellt sind, erscheinen mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 394: Serielle Schnittstellen mit Standard Datenformat

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	C/S	Datenbyte	Steuer-/Statusbyte
	D2	D1	Datenbytes	
	D4	D3		

Diese Sonderklemmen stellen sich jeweils mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.5 Datenaustauschklemmen

750-654, (und die Variante /000-001)

Die Datenaustauschklemmen erscheinen mit jeweils insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 395: Datenaustauschklemmen 750-654, /000-001

Ein- und Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Datenbytes
n+1	D3	D2	

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.6 SSI-Geber Interface Busklemmen

750-630, (und die Varianten /000-001, -002, -006, -008, -009, -011, -012, -013)

Hinweis

Das Prozessabbild der /003-000-Varianten ist abhängig von der parametrierten Betriebart!
Bei den frei parametrierbaren Busklemmenvarianten /003-000 kann die gewünschte Betriebsart eingestellt werden. Abhängig davon, ist das Prozessabbild dieser Busklemmen dann das gleiche, wie das von der entsprechenden Variante.

Die SSI-Geber Interface Busklemmen mit Status erscheinen mit insgesamt 4 Datenbytes im Eingangsbereich des Prozessabbilds. Dabei werden mit word-alignment insgesamt 2 Worte im Prozessabbild belegt.

Tabelle 396: SSI-Geber Interface Busklemmen mit alternativem Datenformat 750-630, /000-0xx

Eingangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	D1	D0	Datenbytes
n+1	D3	D2	

Diese Sonderklemmen stellen sich mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen.

750-630/000-004, -005, -007

Die SSI-Geber Interface Busklemmen mit Status erscheinen mit insgesamt 5 Bytes Nutzdaten im Eingangsprozessabbild, 4 Datenbytes und ein zusätzliches Statusbyte. Dabei werden mit word-alignment insgesamt 3 Worte im Prozessabbild belegt.

Tabelle 397: SSI-Geber Interface Busklemmen mit alternativem Datenformat 750-630/000-00x

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Datenbytes	
	D3	D2		

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

13.3.5.7 Weg- und Winkelmessung

750-631/000-004, -010, -011

Die Busklemme erscheint mit 5 Bytes im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 398: Inkremental Encoder Interface Busklemme 750-631/000-004, --010, -011

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Zählerwort	
	-	-	nicht genutzt	
	D4	D3	Latchwort	

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C	Steuerbyte von Zähler 1
	D1	D0	Zählersetzwert von Zähler 1
	-	-	nicht genutzt
	-	-	nicht genutzt

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-634

Die Busklemme 750-634 erscheint mit 5 Bytes (in der Betriebsart Periodendauermessung mit 6 Bytes) im Eingangs- und mit 3 Bytes im Ausgangsbereich des Prozessabbilds. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 399: Inkremental Encoder Interface Busklemme 750-634

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	S	nicht genutzt	Statusbyte
	D1	D0	Zählerwort	
	-	(D2) *)	nicht genutzt	(Periodendauer)
	D4	D3	Latchwort	
*) Ist durch das Steuerbyte die Betriebsart Periodendauermessung eingestellt, wird in D2 zusammen mit D3/D4 die Periodendauer als 24-Bit-Wert ausgegeben.				

Diese Sonderklemmen stellen sich mit 1x8 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C	nicht genutzt	Steuerbyte
	D1	D0	Zählersetzwort	
	-	-	nicht genutzt	
	-	-		

Diese Sonderklemmen stellen sich mit 1x8 Byte dar und belegen in Klasse (0x68) 1 Instanz.

750-637

Die Inkremental Encoder Interface Busklemme erscheint mit 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes und zwei zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 400: Inkremental Encoder Interface Busklemme 750-637

Ein- und Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	
n	-	C0/S0	Steuer-/Statusbyte von Kanal 1
	D1	D0	Datenwerte von Kanal 1
n+1	-	C1/S1	Steuer-/Statusbyte von Kanal 2
	D3	D2	Datenwerte von Kanal 2

Diese Sonderklemmen stellen sich jeweils mit 2x3 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

750-635,
753-635

Die Digitale Impuls Schnittstelle erscheint mit insgesamt 4 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 401: Digitale Impuls Schnittstelle 750-635

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	C0/S0	Datenbyte	Steuer-/Statusbyte
	D2	D1	Datenbytes	

Diese Sonderklemmen stellen sich jeweils mit 1x4 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.8 DC-Drive Controller

750-636

Der DC-Drive Controller 750-636 stellt dem Feldbuskoppler/-controller über 1 logischen Kanal 6 Byte Ein- und Ausgangsprozessabbild zur Verfügung. Die zu sendenden und zu empfangenden Positionsdaten werden in 4 Ausgangsbytes (D0 ... D3) und 4 Eingangsbytes (D0 ... D3) abgelegt. 2 Steuerbytes (C0, C1) und 2 Statusbytes (S0, S1) dienen zur Steuerung der Busklemme und des Antriebs. Alternativ zu den Positionsdaten im Eingangsprozessabbild (D0 ... D3) können erweiterte Statusinformationen (S2 ... S5) eingeblendet werden. Die 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen den Prozessdaten und den erweiterten Statusbytes im Eingangsprozessabbild erfolgt über Bit 3 (ExtendedInfo_ON) im Controlbyte C1 (C1.3). Mit Bit 3 des Statusbytes S1 (S1.3) wird die Umschaltung quittiert.

Tabelle 402: Antriebssteuerung 750-636

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	S1	S0	Status S1	Statusbyte S0
	D1*) / S3**)	D0*) / S2**)	Istposition*) / Erweitertes Statusbyte S3**)	Istposition (LSB)*) / Erweitertes Statusbyte S2**)
	D3*) / S5**)	D2*) / S4**)	Istposition (MSB)*) / Erweitertes Statusbyte S3**)	Istposition*) / Erweitertes Statusbyte S4**)
*) ExtendedInfo_ON = ‘0’. **) ExtendedInfo_ON = ‘1’.				

Ausgangsprozessabbild			
Instanz	**Bezeichnung der Bytes**		**Bemerkung**
	High Byte	**Low Byte**	

n	C1	C0	Steuerbyte C1	Steuerbyte C0
	D1	D0	Sollposition	Sollposition (LSB)
	D3	D2	Sollposition (MSB)	Sollposition

Diese Sonderklemmen stellen sich jeweils mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.9 Steppercontroller

750-670

Der Steppercontroller RS 422 / 24 V / 20 mA 750-670 stellt dem Feldbuskoppler/-controller über 1 logischen Kanal 12 Byte Ein- und Ausgangsprozessabbild zur Verfügung.
Die zu sendenden und zu empfangenden Daten werden in Abhängigkeit von der Betriebsart in bis zu 7 Ausgangsbytes (D0 ... D6) und 7 Eingangsbytes (D0 ... D6) abgelegt. Das Ausgangsbyte D0 und das Eingangsbyte D0 sind reserviert und ohne Funktion. Ein Klemmenbus-Steuer- und Statusbyte (C0, S0) sowie 3 Steuer- und Statusbytes für die Applikation (C1 ... C3, S1 ... S3) dienen zur Kontrolle des Datenflusses.

Die Umschaltung zwischen beiden Prozessabbildern erfolgt über das Bit 5 im Controlbyte C0 (C0.5). Mit dem Bit 5 des Statusbytes S0 (S0.5) wird das Einschalten der Mailbox quittiert.

Tabelle 403: Steppercontroller RS 422 / 24 V / 20 mA 750-670

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	Reserviert	S0	Reserviert	Statusbyte S0
	D1	D0	Prozessdaten*) / Mailbox**)	
	D3	D2		
	D5	D4		
	S3	D6	Statusbyte S3	Prozessdaten*) / Reserviert**)
	S1	S2	Statusbyte S1	Statusbyte S2
*) Zyklisches Prozessabbild (Mailbox ausgeschaltet). **) Mailboxprozessabbild (Mailbox eingeschaltet)				

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	Reserviert	C0	Reserviert	Controlbyte C0
	D1	D0	Prozessdaten*) / Mailbox**)	
	D3	D2		
	D5	D4		
	C3	D6	Controlbyte C3	Prozessdaten*) / Reserviert**)
	C1	C2	Controlbyte C1	Controlbyte C2

*) Zyklisches Prozessabbild (Mailbox ausgeschaltet).
**) Mailboxprozessabbild (Mailbox eingeschaltet)

Diese Sonderklemmen stellen sich jeweils mit 1x12 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.10 RTC-Modul

750-640

Das RTC-Modul erscheint mit insgesamt 6 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 4 Datenbytes, ein zusätzliches Steuer-/Statusbyte und jeweils ein Befehlsbyte (ID). Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 404: RTC-Modul 750-640

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	ID	C/S	Befehlsbyte	Steuer-/Statusbyte
	D1	D0	Datenbytes	
	D3	D2		

Diese Sonderklemmen stellen sich jeweils mit 1x 6 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.11 DALI/DSI-Masterklemme

750-641

Die DALI/DSI-Masterklemme erscheint mit insgesamt 6 Datenbytes im Ein- und Ausgangsbereich des Prozessabbilds, 5 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Dabei werden mit word-alignment jeweils 3 Worte im Prozessabbild belegt.

Tabelle 405: DALI/DSI-Masterklemme 750-641

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	S	DALI-Antwort	Statusbyte
	D2	D1	Message 3	DALI-Adresse
	D4	D3	Message 1	Message 2

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x67) 1 Instanz.

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	C	DALI-Befehl, DSI-Dimmwert	Steuerbyte
	D2	D1	Parameter 2	DALI-Adresse
	D4	D3	Command-Extension	Parameter 1

Diese Sonderklemmen stellen sich mit 1x6 Byte dar und belegen in Klasse (0x68) 1 Instanz.

13.3.5.12 Funkreceiver EnOcean

750-642

Die EnOcean Funkreceiverklemme erscheint mit insgesamt 4 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 3 Datenbytes und ein zusätzliches Steuer-/Statusbyte. Die 3 Bytes Ausgangsdaten werden jedoch nicht genutzt. Dabei werden mit word-alignment jeweils 2 Worte im Prozessabbild belegt.

Tabelle 406: Funkreceiver EnOcean 750-642

Eingangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	D0	S	Datenbyte	Statusbyte
n+1	D2	D1	Datenbytes	

Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C	nicht genutzt	Steuerbyte
n+1	-	-	nicht genutzt	

Diese Sonderklemmen stellen sich jeweils mit 2x2 Byte dar und belegen in Klasse (0x67) 2 Instanzen und in Klasse (0x68) 2 Instanzen.

13.3.5.13 MP-Bus-Masterklemme

750-643

Die MP-Bus-Masterklemme erscheint mit insgesamt 8 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 6 Datenbytes und zwei zusätzliche

Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 4 Worte im Prozessabbild belegt.

Tabelle 407: MP-Bus-Masterklemme 750-643

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	C1/S1	C0/S0	erweitertes Steuer-/Statusbyte	Steuer-/Statusbyte
	D1	D0	Datenbytes	
	D3	D2		
	D5	D4		

Diese Sonderklemmen stellen sich jeweils mit 1x8 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.5.14 *Bluetooth*® RF-Transceiver

750-644

Die Größe des Prozessabbildes der *Bluetooth*®-Busklemme ist in den festgelegten Größen 12, 24 oder 48 Byte einstellbar.
Es besteht aus einem Steuerbyte (Eingang) bzw. Statusbyte (Ausgang), einem Leerbyte, einer 6, 12 oder 18 Byte großen, überlagerbaren Mailbox (Modus 2) und den *Bluetooth*®-Prozessdaten in einem Umfang von 4 bis 46 Byte.

Die *Bluetooth*®-Busklemme belegt also jeweils 12 bis maximal 48 Bytes im Prozessabbild, wobei die Größen des Eingangs- und Ausgangsprozessabbildes stets übereinstimmen.

Das erste Byte enthält das Steuer-/Statusbyte, das zweite ein Leerbyte.
Daran schließen sich bei ausgeblendeter Mailbox unmittelbar Prozessdaten an.
Bei eingeblendeter Mailbox werden je nach deren Größe die ersten 6, 12 oder 18 Byte Prozessdaten von Mailbox-Daten überlagert. Die Bytes im Bereich hinter der optional einblendbaren Mailbox enthalten grundsätzlich Prozessdaten. Den internen Aufbau der *Bluetooth*®-Prozessdaten entnehmen Sie der Dokumentation des *Bluetooth*® RF-Transceivers 750-644.

Die Einstellung der Mailbox- und Prozessabbildgrößen erfolgt mit dem Inbetriebnahmetool WAGO-I/O-*CHECK*.

Tabelle 408: Bluetooth® RF-Transceiver 750-644

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
	D1	D0	Mailbox (0, 3, 6 oder 9 Worte) sowie Prozessdaten (2-23 Worte)	
	D3	D2		
	D5	D4		
	...	...		
	D45	D44		

Die 750-644 stellt sich als Sonderklemme dar. Ihre Prozessdaten (12, 24 oder 48 Byte) belegen je eine Instanz in Klasse 0x67 und 0x68.

13.3.5.15 Schwingstärke/Wälzlagerüberwachung VIB I/O

750-645

Die Schwingstärke/Wälzlagerüberwachung VIB I/O erscheint mit insgesamt 12 Bytes Nutzdaten im Ein- und Ausgangsbereich des Prozessabbilds, 8 Datenbytes und vier zusätzliche Steuer-/Statusbytes. Dabei werden mit word-alignment jeweils 8 Worte im Prozessabbild belegt.

Tabelle 409: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C0/S0	nicht genutzt	Steuer-/Statusbyte (log. Kanal 1, Sensoreingang 1)
	D1	D0	Datenbytes (log. Kanal 1, Sensoreingang 1)	
n+1	-	C1/S1	nicht genutzt	Steuer-/Statusbyte (log. Kanal 2, Sensoreingang 2)
	D3	D2	Datenbytes (log. Kanal 2, Sensoreingang 2)	
n+2	-	C2/S2	nicht genutzt	Steuer-/Statusbyte (log. Kanal 3, Sensoreingang 3)
	D5	D4	Datenbytes (log. Kanal 3, Sensoreingang 3)	
n+3	-	C3/S3	nicht genutzt	Steuer-/Statusbyte (log. Kanal 4, Sensoreingang 4)
	D7	D6	Datenbytes (log. Kanal 4, Sensoreingang 4)	

Diese Sonderklemmen stellen sich jeweils mit 4x3 Byte dar und belegen in Klasse (0x67) 4 Instanzen und in Klasse (0x68) 4 Instanzen.

13.3.5.16 AS-interface Masterklemme

750-655

Das Prozessabbild der AS-interface Masterklemme ist in seiner Länge einstellbar in den festgelegten Größen von 12, 20, 24, 32, 40 oder 48 Byte.
Es besteht aus einem Control- bzw. Statusbyte, einer 0, 6, 10, 12 oder 18 Byte großen Mailbox und den AS-interface Prozessdaten in einem Umfang von 0 bis 32 Byte.
Mit word-alignment belegt die AS-interface Masterklemme also jeweils 6 bis maximal 24 Worte im Prozessabbild.

Das erste Ein- bzw. Ausgangswort enthält das Status- bzw. Controlbyte sowie ein Leerbyte.
Daran schließen sich für die fest eingeblendete Mailbox (Modus 1) die Worte mit Mailboxdaten an.
Wenn die Mailbox überlagerbar eingestellt ist (Modus 2), enthalten diese Worte Mailbox- oder Prozessdaten.
Die weiteren Worte enthalten die restlichen Prozessdaten.

Tabelle 410: AS-interface Masterklemme 750-655

Ein- und Ausgangsprozessabbild				
Instanz	**Bezeichnung der Bytes**		**Bemerkung**	
	High Byte	**Low Byte**		
n	-	C0/S0	nicht genutzt	Steuer-/Statusbyte
	D1	D0	Mailbox (0,3,5,6 oder 9 Worte) / Prozessdaten (0-16 Worte)	
	D3	D2		
	D5	D4		
	...	...		
	D45	D44		

Diese Sonderklemmen stellen sich jeweils mit 1x 12...48 Byte dar und belegen in Klasse (0x67) 1 Instanz und in Klasse (0x68) 1 Instanz.

13.3.6 Systemklemmen

13.3.6.1 Systemklemmen mit Diagnose

750-610, -611

Die Potentialeinspeiseklemmen 750-610 und –611 mit Diagnose liefern zur Überwachung der Versorgung 2 Bits in das Prozesseingangsabbild.

Tabelle 411: Systemklemmen mit Diagnose 750-610, -611

Eingangsprozessabbild							
Bit 7	**Bit 6**	**Bit 5**	**Bit 4**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
						Diagnosebit S 2 Sicherung	Diagnosebit S 1 Spannung

Die Eingangsklemmen belegen in Klasse (0x65) 2 Instanzen.

13.3.6.2 Binäre Platzhalterklemme

750-622

Die binären Platzhalterklemmen 750-622 verhalten sich wahlweise wie 2 Kanal Digitaleingangsklemmen oder -ausgangsklemmen und belegen je nach angewählter Einstellung pro Kanal 1, 2, 3 oder 4 Bits.

Dabei werden dann entsprechend 2, 4, 6 oder 8 Bits entweder im Prozesseingangs- oder -ausgangsabbild belegt.

Tabelle 412: Binäre Platzhalterklemmen 750-622

Ein- oder Ausgangsprozessabbild							
Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
(Datenbit DI 8)	(Datenbit DI 7)	(Datenbit DI 6)	(Datenbit DI 5)	(Datenbit DI 4)	(Datenbit DI 3)	Datenbit DI 2	Datenbit DI 1

Die Platzhalterklemmen belegen in Klasse (0x65) bzw. in Klasse (0x66) 2, 4, 6 oder 8 Instanzen.

14 Anwendungsbeispiele

14.1 Test von MODBUS-Protokoll und Feldbusknoten

Zum Testen der Funktion Ihres Feldbusknotens benötigen Sie einen MODBUS-Master. Hierfür werden unterschiedliche PC-Applikationen von diversen Herstellern angeboten, die Sie zum Teil als kostenfreie Demoversionen aus dem Internet herunterladen können.

Eines der Programme zum Test Ihres ETHERNET-Feldbusknotens ist **ModScan** der Firma Win-Tech.

Information

Weitere Information
Eine kostenlose Demoversion des Programmes ModScan32 sowie weitere Zusatzprogramme der Firma Win-Tech finden Sie im Internet unter: http://www.win-tech.com/html/demos.htm

ModScan32 ist eine Windows-Applikation, die als MODBUS-Master arbeitet.

Mit diesem Programm können Sie auf die Datenpunkte Ihres angeschlossenen ETHERNET-TCP/IP-Feldbusknotens zugreifen und gewünschte Änderungen vornehmen.

Information

Weitere Information
Eine Beispiel-Beschreibung zur Software-Bedienung finden Sie im Internet unter: http://www.win-tech.com/html/modscan32.htm

14.2 Visualisierung und Steuerung mittels SCADA-Software

Dieses Kapitel vermittelt Ihnen einen kurzen Einblick zum Einsatz des (programmierbaren) WAGO-ETHERNET-Feldbuskopplers/-controllers mit einer Standard-Anwendersoftware zur Prozessvisualisierung und -steuerung.

Das Angebot an Prozessvisualisierungsprogrammen diverser Hersteller, sogenannte SCADA-Software, ist vielfältig.

SCADA ist die Abkürzung für „Supervisory Control and Data Acquisition“ und umfasst Fernwirk- und Datenerfassungssysteme.

Dabei handelt es sich um produktionsnahe, bedienerorientierte Werkzeuge, die als Produktionsinformationssysteme für die Bereiche Automatisierungstechnik, Prozesssteuerung und Produktionsüberwachung genutzt werden.

Der Einsatz von SCADA-Systemen umfasst die Bereiche Visualisierung und Überwachung, Datenzugriff, Trendaufzeichnung, Ereignis- und

Alarmbearbeitung, Prozessanalyse sowie den gezielten Eingriff in einen Prozess (Steuerung).

Der WAGO-ETHERNET-Feldbusknoten stellt dazu die benötigten Prozesseingangs- und -ausgangswerte bereit.

Hinweis

Nur SCADA-Software mit MODBUS-Unterstützung und MODBUS-Treiber verwenden!
Achten Sie bei der Auswahl einer geeigneten SCADA-Software unbedingt darauf, dass ein MODBUS-Gerätetreiber zur Verfügung steht und die im Feldbuskoppler/-controller realisierten MODBUS/TCP-Funktionen unterstützt werden.

Visualisierungsprogramme mit MODBUS-Gerätetreiber werden u. a. von den Firmen Wonderware, National Instruments, Think&Do oder KEPware Inc. angeboten und sind teilweise auch als Demoversion im Internet frei erhältlich.

Die Bedienung dieser Programme ist herstellerspezifisch. Dennoch sind im Folgenden einige wesentliche Schritte aufgeführt, die veranschaulichen, wie ein Programm mit einem WAGO-ETHERNET-Feldbusknoten und einer SCADA-Software prinzipiell entwickelt werden kann:

1. Laden Sie zunächst den MODBUS-Treiber und wählen Sie MODBUS-ETHERNET.

2. Geben Sie die IP-Adresse zur Adressierung des Feldbusknotens ein.

In einigen Programmen können zudem Aliasnamen, z. B. „Messdaten“, für einen Knoten vergeben werden. Die Adressierung kann dann über diesen Namen erfolgen.

3. Kreieren Sie ein grafisches Objekt, wie beispielsweise einen Schalter (digital) oder ein Potenziometer (analog).

Das kreierte Objekt wird auf der Benutzeroberfläche dargestellt.

4. Verknüpfen Sie das Objekt mit dem gewünschten Datenpunkt an dem Knoten, indem Sie folgende Daten eingeben:

   - Knotenadresse (IP-Adresse oder Aliasnamen)
   - Gewünschter MODBUS-Funktionscode (Register/Bit lesen/schreiben)
   - MODBUS-Adresse des gewählten Kanals

Die Eingabe erfolgt programmspezifisch.

Die MODBUS-Adresse eines Busklemmenkanals enthält je nach Anwendersoftware bis zu 5 Stellen.

Beispiel einer MODBUS-Adressierung

Bei der SCADA-Software Lookout der Firma National Instruments werden 6-stellige MODBUS-Adressen verwendet.

Dabei repräsentiert die erste Stelle die MODBUS-Tabelle (0, 1, 3 oder 4) und implizit den Funktionscode (siehe nachfolgende Tabelle).

Tabelle 413: MODBUS-Tabelle und -Funktionscodes

MODBUS-Tabelle	MODBUS-Funktionscode	
0	FC1 oder FC15	Lesen eines Eingangsbits oder Schreiben mehrerer Ausgangsbits
1	FC2	Lesen mehrerer Eingangsbits
3	FC4	Lesen mehrerer Eingangsregistern
4	FC3 oder FC 16	Lesen mehrerer Eingangsregistern oder Schreiben mehrerer Ausgangsregister

Die folgenden fünf Stellen geben die Kanalnummer (beginnend mit 1) der durchnummerierten digitalen oder analogen Eingangs- oder Ausgangskanäle an.

Beispiele:

- Lesen/Schreiben des ersten digitalen Einganges: z. B. 0 0000 1
- Lesen/Schreiben des zweiten analogen Einganges: z. B. 3 0000 2

Anwendungsbeispiel:

Mit der Eingabe: „Messdaten . 0 0000 2“ kann beispielsweise der digitale Eingangskanal 2 des o. g. Knotens „Messdaten“ ausgelesen werden.

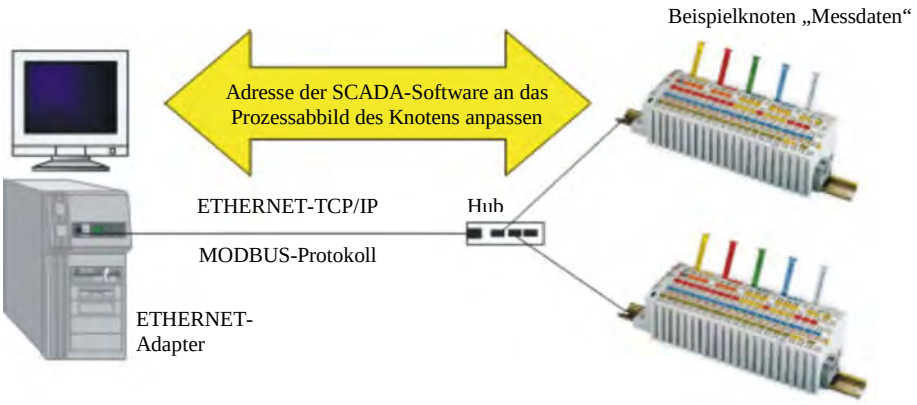


Abbildung 73: Beispiel SCADA-Software mit MODBUS-Treiber

Information

Weitere Information

Eine detaillierte Beschreibung der jeweiligen Software-Bedienung entnehmen Sie dem Handbuch des entsprechenden SCADA-Produktes.

15 Einsatz in explosionsgefährdeten Bereichen

Das **WAGO-I/O-SYSTEM 750** (elektrische Betriebsmittel) ist für den Einsatz in explosionsgefährdeten Bereichen der Zone 2 ausgelegt.

Die nachfolgenden Kapitel beinhalten die allgemeine Kennzeichnung der Komponenten sowie die zu berücksichtigenden Errichtungsbestimmungen. Die einzelnen Abschnitte im Kapitel „Errichtungsbestimmungen“ müssen berücksichtigt werden, falls die Busklemme die entsprechende Zulassung besitzt oder dem Anwendungsbereich der ATEX-Richtlinie unterliegt.

15.1 Beispielhafter Aufbau der Kennzeichnung

15.1.1 Kennzeichnung für Europa gemäß ATEX und IEC-Ex

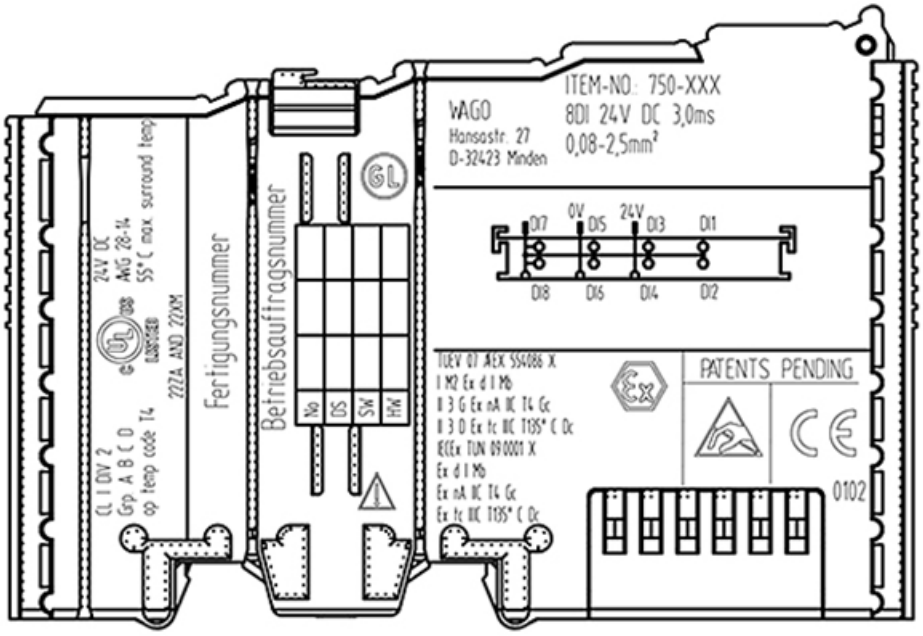


Abbildung 74: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

TUEV 07 ATEX 554086 X
I M2 Ex d I Mb
II 3 G Ex nA IIC T4 Gc
II 3 D Ex tc IIIC T135° C Dc
IECEx TUN 09.0001 X
Ex d I Mb
Ex nA IIC T4 Gc
Ex tc IIIC T135° C Dc

Abbildung 75: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

Tabelle 414: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen.

Bedruckungstext	**Beschreibung**
TÜV 07 ATEX 554086 X IECEx TUN 09.0001 X	Zulassungsbehörde bzw. Bescheinigungsnummern
Stäube	
II	Gerätegruppe: alle außer Bergbau
3D	Gerätekategorie 3 (Zone 22)
Ex	Explosionsschutzkennzeichen
tc Dc	Zündschutzart und Geräteschutzniveau (EPL): Schutz durch Gehäuse
IIIC	Staubgruppe: explosionsfähige Staubatmosphäre
T 135°C	Max. Oberflächentemperatur des Gehäuses (ohne Staubablage)
Bergbau	
I	Gerätegruppe: Bergbau
M2	Gerätekategorie: hohes Maß an Sicherheit
Ex	Explosionsschutzkennzeichen
d Mb	Zündschutzart und Geräteschutzniveau (EPL): druckfeste Kapselung
I	Elektrische Geräte im schlagwettergefährdeten Grubenbau
Gase	
II	Gerätegruppe: Alle außer Bergbau
3G	Gerätekategorie 3 (Zone 2)
Ex	Explosionsschutzkennzeichen
nA Gc	Zündschutzart und Geräteschutzniveau (EPL): nicht funkendes Betriebsmittel
nC Gc	Zündschutzart und Geräteschutzniveau (EPL): funkendes Betriebsmittel, in denen die Kontakte in geeigneter Weise geschützt sind
IIC	Gasgruppe: explosionsfähige Gasatmosphäre
T4	Temperaturklasse: Max. Oberflächentemperatur 135°C

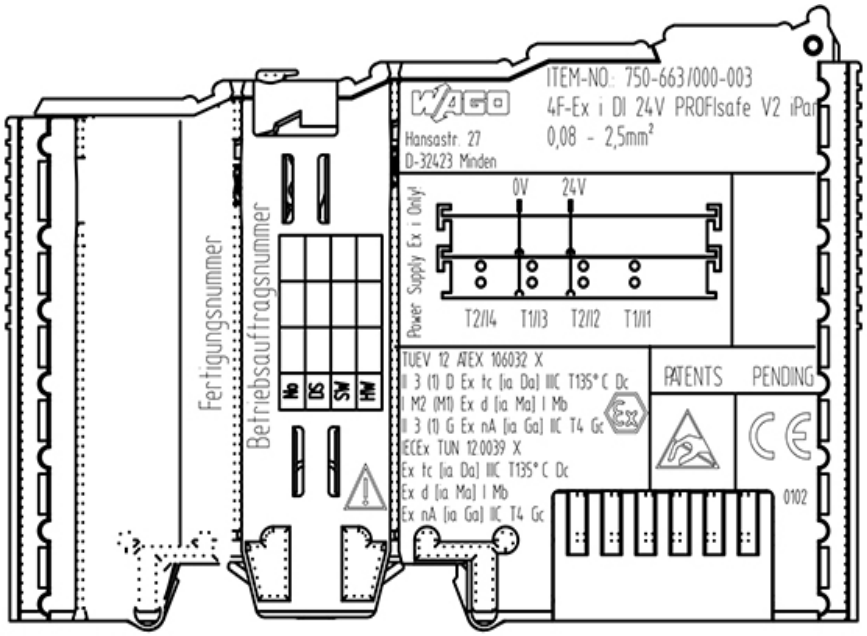


Abbildung 76: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

TUEV 12 ATEX 106032 X
II 3 (1) D Ex tc [ia Da] IIIC T135° C Dc
I M2 (M1) Ex d [ia Ma] I Mb
II 3 (1) G Ex nA [ia Ga] IIC T4 Gc
IECEx TUN 12.0039 X
Ex tc [ia Da] IIIC T135° C Dc
Ex d [ia Ma] I Mb
Ex nA [ia Ga] IIC T4 Gc

Abbildung 77: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Tabelle 415: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Bedruckungstext	**Beschreibung**
TÜV 07 ATEX 554086 X IECEx TUN 09.0001X TÜV 12 ATEX 106032 X IECEx TUN 12.0039 X	Zulassungsbehörde bzw. Bescheinigungsnummern
Stäube	
II	Gerätegruppe: alle außer Bergbau
3(1)D	Gerätekategorie 3 (Zone 22) die Sicherheitsvorrichtungen für Geräte der Kategorie 1 (Zone 20) enthalten
3(2)D	Gerätekategorie 3 (Zone 22) die Sicherheitsvorrichtungen für Geräte der Kategorie 2 (Zone 21) enthalten
Ex	Explosionsschutzkennzeichen
tc Dc	Zündschutzart und Geräteschutzniveau (EPL): Schutz durch Gehäuse
[ia Da]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 20
[ib Db]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 21
IIIC	Staubgruppe: explosionsfähige Staubatmosphäre
T 135°C	Max. Oberflächentemperatur des Gehäuses (ohne Staubablage)
Bergbau	
I	Gerätegruppe: Bergbau
M2 (M1)	Gerätekategorie: hohes Maß an Sicherheit, mit Stromkreisen, die ein sehr hohes Maß an Sicherheit darbieten
Ex d Mb	Explosionsschutzkennzeichen mit Zündschutzart und Geräteschutzniveau (EPL): druckfeste Kapselung
[ia Ma]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen
I	Elektrische Geräte im schlagwettergefährdeten Grubenbau

Tabelle 415: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen.

Gase	
II	Gerätegruppe: Alle außer Bergbau
3(1)G	Gerätekategorie 3 (Zone 2) die Sicherheitsvorrichtungen für Geräte der Kategorie 1 (Zone 0) enthalten
3(2)G	Gerätekategorie 3 (Zone 2) die Sicherheitsvorrichtungen für Geräte der Kategorie 2 (Zone 1) enthalten
Ex	Explosionsschutzkennzeichen
nA Gc	Zündschutzart und Geräteschutzniveau (EPL): nicht funkendes Betriebsmittel
[ia Ga]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 0
[ib Gb]	Zündschutzart und Geräteschutzniveau (EPL): zugehöriges Betriebsmittel mit eigensicheren Stromkreisen für Zone 1
IIC	Gasgruppe: explosionsfähige Gasatmosphäre
T4	Temperaturklasse: Max. Oberflächentemperatur 135°C

15.1.2 Kennzeichnung für Amerika gemäß NEC 500

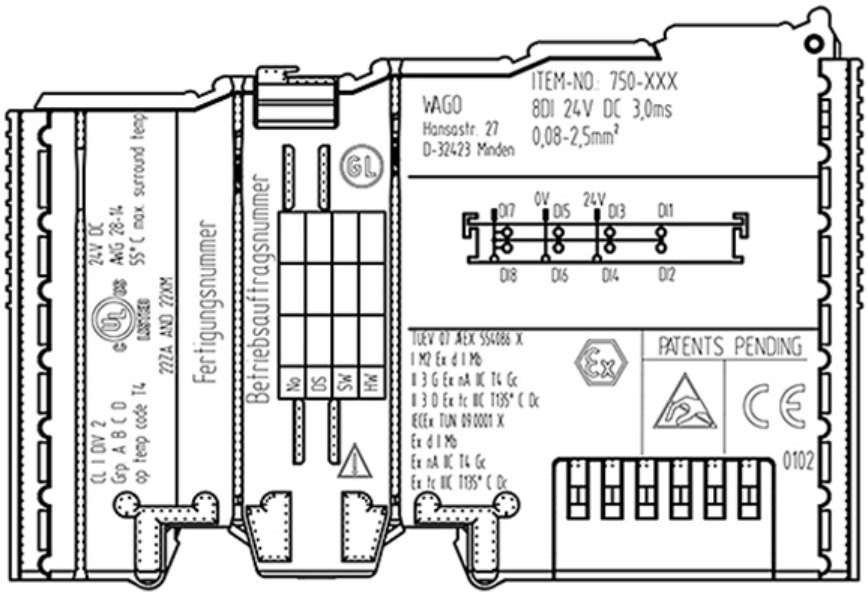


Abbildung 78: Beispiel für seitliche Bedruckung der Busklemmen gemäß NEC 500

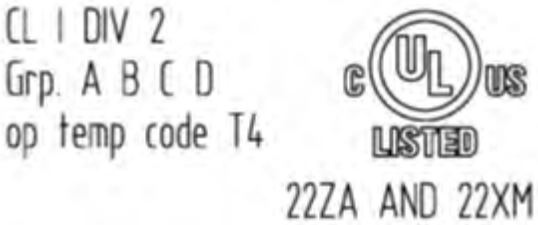


Abbildung 79: Textdetail - Beispielbedruckung der Busklemmen gemäß NEC 500

Tabelle 416: Beschreibung der Beispielbedruckung der Busklemmen gemäß NEC 500

Bedruckungstext	Beschreibung
CL I	Explosionsschutzgruppe (Gefahrenkategorie)
DIV 2	Einsatzbereich
Grp. ABCD	Explosionsgruppe (Gasgruppe)
Op temp code T4	Temperaturklasse

15.2 Errichtungsbestimmungen

Für die Errichtung und den Betrieb elektrischer Anlagen in explosionsfähigen Bereichen sind die am Einsatzort geltenden nationalen und internationalen Bestimmungen und Verordnungen zu beachten.

15.2.1 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 07 ATEX 554086 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) EN 60079-0, EN 60079-11, EN 60079-15 und EN 60079-31 erfüllt.
   Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß EN 60079-0 und EN 60079-1 gewährleistet mit der Schutzart IP64.
   Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExNB bescheinigt sein.
2. Außerhalb des Gerätes sind geeignete Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn eine explosionsfähige Atmosphäre ausgeschlossen werden kann.
4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
   Das ist ebenfalls und im Besonderen gültig für die Schnittstellen „Memory - Card“, „USB“, „Fieldbus connection“, „Configuration and programming interface“, „antenna socket“, „D-Sub“, „DVI-port“ und das „Ethernet interface“. Diese Schnittstellen sind keine energiebegrenzten oder eigensichere Kreise. Die Verwendung dieser Schnittstellen erfolgt in Verantwortung des Betreibers.
5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 und 750-633 muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der EN 60664-1 definiert.
6. Bei austauschbaren Sicherungen ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.
7. In der Nähe des Gerätes sind die folgenden Warnhinweise anzubringen:
   WARNHINWEIS – SICHERUNG NICHT UNTER SPANNUNG HERAUSNEHMEN ODER WECHSELN
   WARNHINWEIS – NICHT UNTER SPANNUNG TRENNEN
   WARNHINWEIS – NUR IN EINEM NICHT EXPLOSIONSGEFÄHRDETEN BEREICH TRENNEN

15.2.2 Besondere Bedingungen für den sicheren Ex Betrieb (ATEX Zertifikat TÜV 12 ATEX 106032 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** Ex i in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) EN 60079-0, EN 60079-11, EN 60079-15 und EN 60079-31 erfüllt.
   Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß EN 60079-0 und EN 60079-1 gewährleistet mit der Schutzart IP64.
   Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExNB bescheinigt sein.
2. Außerhalb des Gerätes sind geeignete Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
4. Für das Gerät muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der EN 60664-1 definiert.

15.2.3 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 09.0001 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGOI/O-SYSTEM 750-*** in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) IEC 60079-0, IEC 60079-11, IEC 60079-15, IEC 60079-31 erfüllt.
   Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß IEC 60079-0 und IEC 60079-1 gewährleistet mit der Schutzart IP64.
   Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExCB bescheinigt sein.
2. Außerhalb des Gerätes sind Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40 % überschritten wird.
3. DIP-Schalter, Kodierschalter und Potentiometer, die an die Busklemme angeschlossen sind, dürfen nur betätigt werden, wenn eine explosionsfähige Atmosphäre ausgeschlossen werden kann.
4. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
   Das ist ebenfalls und im Besonderen gültig für die Schnittstellen „Memory-Card“, „USB“, „Fieldbus connection“, „Configuration and programming interface“, „antenna socket“, „D-Sub“, „DVI-port“ und das „Ethernet interface“. Diese Schnittstellen sind keine energiebegrenzten oder eigensicheren Kreise. Die Verwendung dieser Schnittstellen erfolgt in Verantwortung des Betreibers.
5. Für die Typen 750-606, 750-625/000-001, 750-487/003-000, 750-484 und 750-633 muss Folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert.
6. Bei austauschbaren Sicherungen ist Folgendes zu berücksichtigen: Die Sicherung darf nicht entfernt oder getauscht werden, wenn das Gerät in Betrieb ist.
7. In der Nähe des Gerätes sind die folgenden Warnhinweise anzubringen:
   WARNHINWEIS – SICHERUNG NICHT UNTER SPANNUNG HERAUSNEHMEN ODER WECHSELN
   WARNHINWEIS – NICHT UNTER SPANNUNG TRENNEN
   WARNHINWEIS – NUR IN EINEM NICHT EXPLOSIONSGEFÄHRDETEN BEREICH TRENNEN

15.2.4 Besondere Bedingungen für den sicheren Ex Betrieb (IEC-Ex Zertifikat IECEx TUN 12.0039 X)

1. Für den Betrieb als Gc- oder Dc-Gerät (in Zone 2 oder 22) ist das WAGO-I/O-SYSTEM 750-*** Ex i in einem Gehäuse zu errichten, das die Anforderungen an ein Gerät nach den zutreffenden Normen (siehe Kennzeichnung) IEC 60079-0, IEC 60079-11, IEC 60079-15 und IEC 60079-31 erfüllt.
   Für den Betrieb als Gerät der Gruppe I, Kategorie M2, ist das Gerät in einem Gehäuse zu errichten, das einen ausreichenden Schutz gemäß IEC 60079-0 und IEC 60079-1 gewährleistet mit der Schutzart IP64.
   Die Übereinstimmung mit diesen Anforderungen und dem korrekten Einbau des Gerätes in ein Gehäuse oder Schaltschrank muss durch einen ExCB bescheinigt sein.
2. Außerhalb des Gerätes sind Maßnahmen zu treffen, sodass die Bemessungsspannung durch vorübergehende Störungen um nicht mehr als 40% überschritten wird.
3. Das Anschließen und Abklemmen von nicht eigensicheren Stromkreisen ist nur zulässig für die Installation, die Wartung und die Reparatur. Das zeitliche Zusammentreffen von explosiver Atmosphäre und der Installation, der Wartung und der Reparatur muss ausgeschlossen werden.
4. Für das Gerät muss folgendes berücksichtigt werden: Die Schnittstellenstromkreise müssen begrenzt werden auf die Überspannungskategorie I/II/III (Stromkreise ohne Netzversorgung/ Stromkreise mit Netzversorgung) wie in der IEC 60664-1 definiert.

15.2.5 Besondere Bedingungen für den sicheren Betrieb nach ANSI/ISA 12.12.01

A. „Dieses Gerät ist ausschließlich für den Einsatz in Class I, Division 2, Gruppen A, B, C, D oder nicht explosionsgefährdeten Bereichen geeignet."
B. „Dieses Gerät muss in werkzeuggesicherte Gehäuse eingebaut werden."
C. „WARNUNG - Explosionsgefahr - Der Austausch von Komponenten kann die Eignung für Class I, Division 2 beeinträchtigen."
D. „WARNUNG - Klemmen Sie das Gerät nur dann ab, wenn die Versorgung ausgeschaltet ist oder wenn der Bereich als nicht-explosionsgefährdet gilt" ist in der Nähe von Steckverbindern und Sicherungshaltern anzubringen, die für Bediener zugänglich sind.
E. Falls eine Sicherung vorhanden ist, muss folgende Information vorhanden sein: „Es muss ein Schalter vorgesehen sein, der für den Einsatzort geeignet ist, in dem das Gerät installiert wird, um die Sicherung von der Versorgung zu trennen."
F. Für Baugruppen mit EtherCAT/Ethernet-Steckverbindern gilt: „Nur für den Einsatz in LAN, nicht für den Anschluss an Fernmeldeleitungen."
G. „WARNUNG - Die Klemme 750-642 ist nur mit dem Antennenmodul 758-910 zu benutzen."
H. Für Feldbuskoppler/-controller und Economy-Busmodule gilt: „Die Service-Schnittstelle ist nur für einen vorübergehenden Anschluss bestimmt. Verbinden oder trennen sie diese nur, wenn der Bereich als nicht-explosionsgefährdet gilt. Das Verbinden oder Trennen in einer explosionsgefährdeten Atmosphäre könnte zu einer Explosion führen."
I. Für Geräte mit Sicherung gilt: „WARNUNG - Geräte mit Sicherungen dürfen nicht in Stromkreise integriert werden, die einer Überlast ausgesetzt sind, z.B. Motorkreise."
J. Für Geräte mit SD-Karte gilt: „WARNUNG - Stecken oder ziehen Sie die SD-Karte bei anliegender Spannung nur dann, wenn es sicher ist, dass der Bereich frei von zündfähigen Gasen oder Dämpfen ist."

Information

Weitere Information

Einen Zertifizierungsnachweis erhalten Sie auf Anfrage. Beachten Sie auch die Hinweise auf dem Beipackzettel der Busklemme. Das Handbuch mit den oben aufgeführten Bedingungen für sicheren Gebrauch muss für den Anwender jederzeit zur Verfügung stehen.

Abbildungsverzeichnis

Abbildung 1: Feldbusknoten (Beispiel) 19
Abbildung 1: Beispiel einer seitlichen Gehäusebedruckung 20
Abbildung 2: Beispiel einer Fertigungsnummer 20
Abbildung 1: Potentialtrennung (Beispiel) 24
Abbildung 1: Systemversorgung 25
Abbildung 1: Systemspannung (Beispiel) 26
Abbildung 1: Feldversorgung (Sensor/Aktor) für ECO-Feldbuskoppler 29
Abbildung 1: Potentialeinspeiseklemme mit Sicherungshalter (Beispiel 750-610) 31
Abbildung 2: Sicherungshalter ziehen 32
Abbildung 3: Sicherungshalter öffnen 32
Abbildung 4: Sicherung wechseln 32
Abbildung 5: Sicherungsklemmen für Kfz-Sicherungen, Serie 282 33
Abbildung 6: Sicherungsklemmen für Kfz-Sicherungen, Serie 2006 33
Abbildung 7: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 281 33
Abbildung 8: Sicherungsklemmen mit schwenkbarem Sicherungshalter, Serie 2002 33
Abbildung 1: Versorgungsbeispiel 34
Abbildung 1: Tragschienenkontakt (Beispiel) 38
Abbildung 1: Kabelschirm auf Erdpotential 39
Abbildung 2: Beispiel WAGO-Schirm-Anschlusssystem 40
Abbildung 3: Anwendung des WAGO-Schirm-Anschlusssystems 41
Abbildung 1: Ansicht ETHERNET TCP/IP-Feldbuscontroller 45
Abbildung 1: Geräteeinspeisung 47
Abbildung 1: RJ-45-Stecker 48
Abbildung 1: Anzeigeelemente 49
Abbildung 1: Service-Schnittstelle (geschlossene und geöffnete Klappe) 50
Abbildung 1: Betriebsartenschalter (geschlossene und geöffnete Klappe der Service-Schnittstelle) 51
Abbildung 1: Adresswahlschalter (hier Einstellung „0“) 53
Abbildung 1: Abstände 62
Abbildung 1: Verriegelung 65
Abbildung 1: Busklemme einsetzen (Beispiel) 66
Abbildung 2: Busklemme einrasten (Beispiel) 66
Abbildung 1: Busklemme entfernen (Beispiel) 67
Abbildung 1: Datenkontakte 68
Abbildung 1: Beispiele für die Anordnung von Leistungskontakten 69
Abbildung 1: Leiter an CAGE CLAMP® anschließen 70
Abbildung 1: Anlauf des Feldbuscontrollers 72
Abbildung 1: Beispiel Eingangsprozessabbild 75
Abbildung 2: Beispiel Ausgangsprozessabbild 76
Abbildung 1: Speicherbereiche und Datenaustausch 80
Abbildung 1: Beispiel Deklarierung für remanente Merker unter „VAR RETAIN“ 82
Abbildung 1: Datenaustausch zwischen MODBUS-Master und Busklemmen 89

Abbildung 1: Datenaustausch zwischen SPS-Funktionalität (CPU) des PFCs und Busklemmen 91
Abbildung 1: Adressierungsbeispiel für einen Feldbusknoten 94
Abbildung 1: Adresswahlschalter, hier z. B. Einstellung des Wertes „50“ ($2^1 + 2^4 + 2^5$) 98
Abbildung 1: Beispiel für den Funktionstest eines Feldbusknotens 111
Abbildung 1: ETHERNET-Settings-Beispiel für die Zeitsynchronisation 116
Abbildung 1: Dialogfenster Zielsystemeinstellungen 120
Abbildung 1: Konfigurationsdatei „EA-config.xml“ 125
Abbildung 1: Watchdog-Laufzeit kleiner als Tasklaufzeit 131
Abbildung 2: Watchdog-Laufzeit größer als Task-Aufrufintervall 131
Abbildung 1: Systemereignisse aktivieren/deaktivieren 134
Abbildung 1: Dialogfenster „Kommunikationsparameter“, Erstellen einer neuen Verbindung 138
Abbildung 1: Anmeldung für Programmierzugriff 139
Abbildung 1: WBM-Seite „Information“ 144
Abbildung 1: WBM-Seite „Ethernet“ 146
Abbildung 1: WBM-Seite „TCP/IP“ 149
Abbildung 1: WBM-Seite „Port“ 151
Abbildung 1: WBM-Seite „Watchdog“ 153
Abbildung 1: WBM-Seite „Clock“ 155
Abbildung 1: WBM-Seite „Security“ 158
Abbildung 1: WBM-Seite „MODBUS“ 160
Abbildung 1: WBM-Seite „EtherNet/IP“ 162
Abbildung 1: WBM-Seite „PLC-Info“ 164
Abbildung 1: WBM-Seite „PLC“ 165
Abbildung 2: WBM-Seite „Features“ 167
Abbildung 1: WBM-Seite „I/O Configuration“ (Beispielkonfiguration) 168
Abbildung 1: WBM-Seite „Disk Info“ 170
Abbildung 1: Anzeigeelemente 171
Abbildung 1: Knotenstatus -Signalisierung der I/O-LED 174
Abbildung 2: Kodierung der Fehlermeldung 174
Abbildung 1: Funktionsblock zur Ermittlung des Feldbusausfalls 182
Abbildung 1: Anwendung von MODBUS-Funktionen für einen Feldbuskoppler/-controller 202
Abbildung 1: Beispiel SCADA-Software mit MODBUS-Treiber 362
Abbildung 1: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Busklemmen. 364
Abbildung 2: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen. 364
Abbildung 1: Beispiel für die seitliche Bedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen. 366
Abbildung 2: Textdetail - Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen. 366
Abbildung 1: Beispiel für seitliche Bedruckung der Busklemmen gemäß NEC 500 369
Abbildung 2: Textdetail - Beispielbedruckung der Busklemmen gemäß NEC 500 369

Tabellenverzeichnis

Tabelle 1: Darstellungen der Zahlensysteme 13
Tabelle 1: Schriftkonventionen 13
Tabelle 1: Auslegung 26
Tabelle 1: Legende zur Abbildung „Feldversorgung (Sensor/Aktor) für ECO-Feldbuskoppler” 29
Tabelle 1: Potentialeinspeiseklemmen 30
Tabelle 1: Legende zur Abbildung „Versorgungsbeispiel für Feldbuskoppler/-controller” 35
Tabelle 1: WAGO-Netzgeräte (Auswahl) 36
Tabelle 1: WAGO-Schutzleiterklemmen 37
Tabelle 1: Legende zur Ansicht „ETHERNET-Feldbuscontroller“ 46
Tabelle 1: Busanschluss und Steckerbelegung, RJ-45-Stecker 48
Tabelle 1: Anzeigeelemente Feldbusstatus 49
Tabelle 1: Anzeigeelemente Knotenstatus 49
Tabelle 1: Legende zur Abbildung „Service-Schnittstelle (geschlossene und geöffnete Klappe)“ 50
Tabelle 1: Legende zur Abbildung „Betriebsartenschalter“ 51
Tabelle 1: Betriebsartenschalterstellungen, statische Positionen bei PowerOn/Reset 52
Tabelle 2: Betriebsartenschalterstellungen, dynamische Positionen im laufenden Betrieb 52
Tabelle 1: Bedeutung Schalterstellungen des DIP-Schalters 53
Tabelle 1: Technische Daten – Gerätedaten 54
Tabelle 2: Technische Daten – Systemdaten 54
Tabelle 3: Technische Daten – Versorgung 55
Tabelle 4: Technische Daten – Feldbus MODBUS/TCP 55
Tabelle 5: Technische Daten – Zubehör 55
Tabelle 11: Technische Daten – Verdrahtungsebene 55
Tabelle 1: Technische Daten – Leistungskontakte 55
Tabelle 1: Technische Daten – Datenkontakte 55
Tabelle 1: Technische Daten – klimatische Umweltbedingungen 56
Tabelle 1: Technische Daten – Mechanische Festigkeit 56
Tabelle 1: WAGO-Tragschienen 62
Tabelle 1: Datenbreite der Busklemmen (Beispiele) 84
Tabelle 1: IEC-61131-3-Adressräume 85
Tabelle 1: Absolute Adressen 86
Tabelle 1: Beispieladressierung 86
Tabelle 1: Zuordnung digitale Ein-/Ausgänge zum Prozessdatenwort gemäß Intel-Format 88
Tabelle 1: ETHERNET-Bibliotheken für WAGO-I/O-*PRO* 127
Tabelle 1: Task-Abarbeitung 132
Tabelle 1: WBM-Seite „Information“ 145
Tabelle 1: WBM-Seite „Ethernet“ 147
Tabelle 1: WBM-Seite „TCP/IP“ 149
Tabelle 1: WBM-Seite „Port“ 152
Tabelle 1: WBM-Seite „Watchdog“ 154
Tabelle 1: WBM-Seite „Clock“ 156
Tabelle 1: WBM-Seite „Security“ 158

Tabelle 1: WBM-Seite „Modbus“ 161
Tabelle 1: WBM-Seite „EtherNet/IP“ 163
Tabelle 1: WBM-Seite „PLC-Info“ 164
Tabelle 1: WBM-Seite „PLC“ 166
Tabelle 1: WBM-Seite „Features“ 167
Tabelle 1: WBM-Seite „I/O Config“ 169
Tabelle 1: WBM-Seite „Disk Info“ 170
Tabelle 1: LED-Zuordnung für die Diagnose 171
Tabelle 1: Diagnose des Feldbusstatus – Abhilfe im Fehlerfall 172
Tabelle 1: Diagnose des Knotenstatus – Abhilfe im Fehlerfall 173
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 1 175
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 2 176
Tabelle 2: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 3 177
Tabelle 3: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 4 178
Tabelle 4: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 5 179
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 6 179
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 7 180
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 8...9180
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 10 .. 180
Tabelle 1: Blinkcode-Tabelle für die I/O-LED-Signalisierung, Fehlercode 11 .. 181
Tabelle 1: IP-Datenpaket 185
Tabelle 2: Netzwerkklasse Class A 186
Tabelle 3: Netzwerkklasse Class B 186
Tabelle 4: Netzwerkklasse Class C 187
Tabelle 5: Eckdaten Class A, B und C 187
Tabelle 6: Beispiel: Klasse B-Adresse mit Feld für Subnetzwerk-ID 188
Tabelle 7: Subnetz-Maske für Class A-Netzwerke 188
Tabelle 8: Subnetz-Maske für Class B-Netzwerke 188
Tabelle 9: Subnetz-Maske für Class C-Netzwerke 188
Tabelle 10: Beispiel für eine IP-Adresse aus einem Class B-Netz 189
Tabelle 1: BootP-Optionen 192
Tabelle 1: DHCP-Optionen 195
Tabelle 1: Bedeutung der SNTP-Parameter 196
Tabelle 1: MODBUS/TCP-Header 199
Tabelle 1: Grunddatentypen des MODBUS-Protokolls 200
Tabelle 1: Auflistung der in dem Feldbuscontroller realisierten MODBUS-Funktionen 200
Tabelle 1: Exception-Codes 203
Tabelle 1: Aufbau des Request für den Funktionscode FC1 204
Tabelle 2: Aufbau der Response für den Funktionscode FC1 204
Tabelle 3: Zuordnung der Eingänge 204
Tabelle 4: Aufbau der Exception für den Funktionscode FC1 205
Tabelle 5: Aufbau des Request für den Funktionscode FC2 206
Tabelle 6: Aufbau der Response für den Funktionscode FC2 206
Tabelle 7: Zuordnung der Eingänge 206
Tabelle 8: Aufbau der Exception für den Funktionscode FC2 207
Tabelle 9: Aufbau des Request für den Funktionscode FC3 208
Tabelle 10: Aufbau der Response für den Funktionscode FC3 208
Tabelle 11: Aufbau der Exception für den Funktionscode FC3 208
Tabelle 12: Aufbau des Request für den Funktionscode FC4 209

Tabelle 13: Aufbau der Response für den Funktionscode FC4 209
Tabelle 14: Aufbau der Exception für den Funktionscode FC4 209
Tabelle 15: Aufbau des Request für den Funktionscode FC5 210
Tabelle 16: Aufbau der Response für den Funktionscode FC5 210
Tabelle 17: Aufbau der Exception für den Funktionscode FC5 210
Tabelle 18: Aufbau des Request für den Funktionscode FC6 211
Tabelle 19: Aufbau der Response für den Funktionscode FC6 211
Tabelle 20: Aufbau der Exception für den Funktionscode FC6 211
Tabelle 1: Aufbau des Request für den Funktionscode FC11 212
Tabelle 2: Aufbau der Response für den Funktionscode FC11 212
Tabelle 3: Aufbau der Exception für den Funktionscode FC11 212
Tabelle 4: Aufbau des Request für den Funktionscode FC15 213
Tabelle 5: Aufbau der Response für den Funktionscode FC15 213
Tabelle 6: Aufbau der Exception für den Funktionscode FC15 214
Tabelle 7: Aufbau des Request für den Funktionscode FC16 215
Tabelle 8: Aufbau der Response für den Funktionscode FC16 215
Tabelle 9: Aufbau der Exception für den Funktionscode FC16 215
Tabelle 1: Aufbau des Request für den Funktionscode FC22 216
Tabelle 2: Aufbau der Response für den Funktionscode FC22 216
Tabelle 3: Aufbau der Exception für den Funktionscode FC22 216
Tabelle 1: Aufbau des Request für den Funktionscode FC23 217
Tabelle 2: Aufbau der Response für den Funktionscode FC23 217
Tabelle 3: Aufbau der Exception für den Funktionscode FC23 217
Tabelle 1: Registerzugriff Lesen (mit FC3, FC4 und FC23) 219
Tabelle 1: Registerzugriff Schreiben (mit FC6, FC16, FC22 und FC23) 220
Tabelle 1: Bitzugriff Lesen (mit FC1 und FC2) 221
Tabelle 1: Bitzugriff Schreiben (mit FC5 und FC15) 221
Tabelle 1: MODBUS-Register 222
Tabelle 1: MODBUS-Register (Fortsetzung) 223
Tabelle 1: Registeradresse 0x1000 224
Tabelle 1: Registeradresse 0x1001 225
Tabelle 2: Registeradresse 0x1002 225
Tabelle 3: Registeradresse 0x1003 225
Tabelle 4: Registeradresse 0x1004 226
Tabelle 5: Registeradresse 0x1005 226
Tabelle 6: Registeradresse 0x1006 226
Tabelle 7: Registeradresse 0x1007 226
Tabelle 8: Registeradresse 0x1008 226
Tabelle 9: Registeradresse 0x1009 227
Tabelle 10: Registeradresse 0x100A 227
Tabelle 1: Watchdog starten 227
Tabelle 1: Registeradresse 0x100B 228
Tabelle 1: Registeradresse 0x1020 229
Tabelle 2: Registeradresse 0x1021 229
Tabelle 1: Registeradresse 0x1022 230
Tabelle 2: Registeradresse 0x1023 230
Tabelle 3: Registeradresse 0x1024 230
Tabelle 4: Registeradresse 0x1025 230
Tabelle 1: Registeradresse 0x1028 230
Tabelle 1: Registeradresse 0x1029 231

Tabelle 1: Registeradresse 0x102A 231
Tabelle 1: Registeradresse 0x102B 231
Tabelle 1: Registeradresse 0x1030 231
Tabelle 2: Registeradresse 0x1031 231
Tabelle 1: Registeradresse 0x1035 232
Tabelle 1: Registeradresse 0x1036 232
Tabelle 1: Registeradresse 0x1037 232
Tabelle 1: Registeradresse 0x1050 232
Tabelle 1: Registeradresse 0x2030 233
Tabelle 1: Registeradresse 0x2031 233
Tabelle 2: Registeradresse 0x2032 234
Tabelle 3: Registeradresse 0x2033 234
Tabelle 1: Registeradresse 0x2040 234
Tabelle 1: Registeradresse 0x2041 234
Tabelle 2: Registeradresse 0x2042 235
Tabelle 3: Registeradresse 0x2043 235
Tabelle 1: Registeradresse 0x2010 236
Tabelle 2: Registeradresse 0x2011 236
Tabelle 3: Registeradresse 0x2012 236
Tabelle 4: Registeradresse 0x2013 236
Tabelle 5: Registeradresse 0x2014 236
Tabelle 1: Registeradresse 0x2020 236
Tabelle 2: Registeradresse 0x2021 237
Tabelle 3: Registeradresse 0x2022 237
Tabelle 4: Registeradresse 0x2023 237
Tabelle 1: Registeradresse 0x2000 238
Tabelle 2: Registeradresse 0x2001 238
Tabelle 3: Registeradresse 0x2002 238
Tabelle 4: Registeradresse 0x2003 238
Tabelle 5: Registeradresse 0x2004 238
Tabelle 6: Registeradresse 0x2005 238
Tabelle 7: Registeradresse 0x2006 239
Tabelle 8: Registeradresse 0x2007 239
Tabelle 9: Registeradresse 0x2008 239
Tabelle 1: Registeradresse 0x3000 bis 0x4FFF 239
Tabelle 1: OSI-Referenzmodell 241
Tabelle 2: Übersicht CIP-Common-Klassen 244
Tabelle 3: Übersicht WAGO-spezifische Klassen 244
Tabelle 1: Erläuterung der Tabellenköpfe in den Objektbeschreibungen 245
Tabelle 2: Identity (01 $_{hex}$) – Klasse 246
Tabelle 3: Identity (01 $_{hex}$) – Instanz 1 247
Tabelle 4: Identity (01 $_{hex}$) – Common Services 247
Tabelle 5: Message Router (02 $_{hex}$) – Klasse 248
Tabelle 6: Message Router (02 $_{hex}$) – Instanz 1 248
Tabelle 7: Message Router (02 $_{hex}$) – Common Services 249
Tabelle 1: Assembly (04 $_{hex}$) – Klasse 249
Tabelle 2: Statische Assembly-Instanzen – Übersicht 249
Tabelle 3: Statische Assembly-Instanzen – Instanz 101 (65 $_{hex}$) 250
Tabelle 4: Statische Assembly-Instanzen – Instanz 102 (66 $_{hex}$) 250
Tabelle 5: Statische Assembly-Instanzen – Instanz 103 (67 $_{hex}$) 250

Tabelle 6: Statische Assembly-Instanzen – Instanz 104 (68 hex) 250
Tabelle 7: Statische Assembly-Instanzen – Instanz 105 (69 hex) 251
Tabelle 8: Statische Assembly-Instanzen – Instanz 106 (6A hex) 251
Tabelle 9: Statische Assembly-Instanzen – Instanz 107 (6B hex) 251
Tabelle 10: Statische Assembly-Instanzen – Instanz 108 (6C hex) 252
Tabelle 11: Statische Assembly-Instanzen – Instanz 109 (6D hex) 252
Tabelle 1: Statische Assembly-Instanzen – Instanz 110 (6E hex) 252
Tabelle 2: Statische Assembly-Instanzen – Instanz 111 (6F hex) 253
Tabelle 1: Statische Assembly-Instanzen – Common Services 253
Tabelle 1: Port Class (F4 hex) – Klasse 254
Tabelle 2: Port Class (F4 hex) – Instanz 1 255
Tabelle 3: Port Class (F4 hex) – Common Services 255
Tabelle 4: TCP/IP Interface (F5 hex) – Klasse 256
Tabelle 5: TCP/IP Interface (F5 hex) – Instanz 1 256
Tabelle 6: TCP/IP Interface (F5 hex) – Common Services 257
Tabelle 7: Ethernet Link (F6 hex) – Klasse 257
Tabelle 1: Ethernet Link (F6 hex) – Instanz 1 258
Tabelle 2: Ethernet Link (F6 hex) – Instanz 2 260
Tabelle 3: Ethernet Link (F6 hex) – Instanz 3 262
Tabelle 4: Ethernet Link (F6 hex) – Common Services 263
Tabelle 1: Coupler/Controller Configuration (64 hex) – Klasse 264
Tabelle 2: Coupler/Controller Configuration (64 hex) – Instanz 1 264
Tabelle 1: Coupler/Controller Configuration (64 hex) – Common service 266
Tabelle 1: Discrete Input Point (65 hex) – Klasse 266
Tabelle 2: Discrete Input Point (65 hex) – Instanz 1...255 266
Tabelle 3: Discrete Input Point (65 hex) – Common service 266
Tabelle 4: Discrete Input Point Extended 1(69 hex,) – Klasse 267
Tabelle 5: Discrete Input Point Extended 1 (69 hex,) – Instanz 256...510 267
Tabelle 6: Discrete Input Point Extended 1 (69 hex) – Common service 267
Tabelle 7: Discrete Input Point Extended 2 (6D hex) – Klasse 267
Tabelle 8: Discrete Input Point Extended 2 (6D hex) – Instanz 511...765 268
Tabelle 9: Discrete Input Point Extended 2 (6D hex) – Common service 268
Tabelle 10: Discrete Input Point Extended 3 (71 hex) – Klasse 269
Tabelle 11: Discrete Input Point Extended 3 (71 hex) – Instanz 766...1020 269
Tabelle 12: Discrete Input Point Extended 3 (71 hex) – Common service 269
Tabelle 13: Discrete Output Point (66 hex) – Klasse 270
Tabelle 14: Discrete Output Point (66 hex) – Instanz 1...255 270
Tabelle 15: Discrete Output Point (66 hex) – Common service 270
Tabelle 16: Discrete Output Point Extended 1 (6A hex) – Klasse 271
Tabelle 17: Discrete Output Point Extended 1 (6A hex) – Instanz 256...510 271
Tabelle 18: Discrete Output Point Extended 1 (6A hex) – Common service 271
Tabelle 19: Discrete Output Point Extended 2 (6E hex) – Klasse 272
Tabelle 20: Discrete Output Point Extended 2 (6E hex) – Instanz 511...765 272
Tabelle 21: Discrete Output Point Extended 2 (6E hex) – Common service 272
Tabelle 22: Discrete Output Point Extended 3 (72 hex) – Klasse 273
Tabelle 23: Discrete Output Point Extended 3 (72 hex) – Instanz 766...1020 273
Tabelle 24: Discrete Output Point Extended 2 (6E hex) – Common service 273
Tabelle 25: Analog Input Point (67 hex) – Klasse 274
Tabelle 26: Analog Input Point (67 hex) – Instanz 1 ... 255 274
Tabelle 27: Analog Input Point (67 hex) – Common service 274

Tabelle 28: Analog Input Point Extended 1 (6B $_{hex}$) – Klasse 275
Tabelle 29: Analog Input Point Extended 1 (6B $_{hex}$) – Instanz 256 ... 510 275
Tabelle 30: Analog Input Point Extended 1 (6B $_{hex}$) – Common service 275
Tabelle 31: Analog Input Point Extended 2 (6F $_{hex}$) – Klasse 276
Tabelle 32: Analog Input Point Extended 2 (6F $_{hex}$) – Instanz 511 ... 765 276
Tabelle 33: Analog Input Point Extended 2 (6F $_{hex}$) – Common service 276
Tabelle 34: Analog Input Point Extended 3 (73 $_{hex}$) – Klasse 277
Tabelle 35: Analog Input Point Extended 3 (73 $_{hex}$) – Instanz 766 ... 1020 277
Tabelle 36: Analog Input Point Extended 3 (73 $_{hex}$) – Common service 277
Tabelle 37: Analog Output Point (68 $_{hex}$) – Klasse 278
Tabelle 38: Analog Output Point (68 $_{hex}$) – Instanz 1...255 278
Tabelle 39: Analog Output Point (68 $_{hex}$) – Common service 278
Tabelle 40: Analog Output Point Extended 1 (6C $_{hex}$) – Klasse 279
Tabelle 41: Analog Output Point Extended 1 (6C $_{hex}$) – Instanz 256...510 279
Tabelle 42: Analog Output Point Extended 1 (6C $_{hex}$) – Common service 279
Tabelle 43: Analog Output Point Extended 2 (70 $_{hex}$) – Klasse 280
Tabelle 44: Analog Output Point Extended 2 (70 $_{hex}$) – Instanz 511...765 280
Tabelle 45: Analog Output Point Extended 2 (70 $_{hex}$) – Common service 280
Tabelle 46: Analog Output Point Extended 3 (74 $_{hex}$) – Klasse 281
Tabelle 47: Analog Output Point Extended 3 (74 $_{hex}$) – Instanz 766...1020 281
Tabelle 48: Analog Output Point Extended 3 (74 $_{hex}$) – Common service 281
Tabelle 49: Module Configuration (80 $_{hex}$) – Klasse 282
Tabelle 50: Module Configuration (80 $_{hex}$) – Instanz 1...255 282
Tabelle 51: Module Configuration (80 $_{hex}$) – Common service 282
Tabelle 52: Module Configuration Extended (81 $_{hex}$) – Klasse 283
Tabelle 53: Module Configuration Extended (81 $_{hex}$) – Instanz 256 283
Tabelle 54: Module Configuration Extended (81 $_{hex}$) – Common service 283
Tabelle 1: Input Fieldbus Variable USINT (A0 $_{hex}$) – Klasse 284
Tabelle 2: Input Fieldbus Variable USINT (A0 $_{hex}$) – Instanz 1...255 284
Tabelle 3: Input fieldbus variable USINT (A0 $_{hex}$) – Common service 284
Tabelle 4: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Klasse 285
Tabelle 5: Input Fieldbus Variable USINT Extended 1 (A1 $_{hex}$) – Instanz 256...510 285
Tabelle 6: Input fieldbus variable USINT Extended 1 (A1 $_{hex}$) – Common service 285
Tabelle 7: Input Fieldbus Variable USINT Extended 2 (A2 $_{hex}$) – Klasse 286
Tabelle 8: Input Fieldbus Variable USINT Extended 2 (A2 $_{hex}$) – Instanz 511...512 286
Tabelle 9: Input fieldbus variable USINT Extended 2 (A2 $_{hex}$) – Common service 286
Tabelle 10: Output Fieldbus Variable USINT (A3 $_{hex}$) – Klasse 287
Tabelle 11: Output Fieldbus Variable USINT (A3 $_{hex}$) – Instanz 1...255 287
Tabelle 12: Output Fieldbus Variable USINT (A3 $_{hex}$) – Common service 287
Tabelle 13: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Klasse 288
Tabelle 14: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Instanz 256...510 288
Tabelle 15: Output Fieldbus Variable USINT Extended 1 (A4 $_{hex}$) – Common service 288
Tabelle 16: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Klasse 289

Tabelle 17: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Instanz 511...512 ... 289
Tabelle 18: Output Fieldbus Variable USINT Extended 2 (A5 $_{hex}$) – Common service ... 289
Tabelle 19: Input Fieldbus Variable UINT (A6 $_{hex}$) – Klasse ... 290
Tabelle 20: Input Fieldbus Variable UINT (A6 $_{hex}$) – Instanz 1...255 ... 290
Tabelle 21: Input Fieldbus Variable UINT (A6 $_{hex}$) – Common service ... 290
Tabelle 22: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Klasse ... 291
Tabelle 23: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Instanz 256 ... 291
Tabelle 24: Input Fieldbus Variable UINT Extended 1 (A7 $_{hex}$) – Common service ... 291
Tabelle 25: Output Fieldbus Variable UINT (A8 $_{hex}$) – Klasse ... 292
Tabelle 26: Output Fieldbus Variable UINT (A8 $_{hex}$) – Instanz 1...255 ... 292
Tabelle 27: Output Fieldbus Variable UINT (A8 $_{hex}$) – Common service ... 292
Tabelle 28: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Klasse ... 293
Tabelle 29: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Instanz 256 293
Tabelle 30: Output Fieldbus Variable UINT Extended 1 (A9 $_{hex}$) – Common service ... 293
Tabelle 31: Input Fieldbus Variable UDINT (AA $_{hex}$) – Klasse ... 294
Tabelle 32: Input Fieldbus Variable UDINT (AA $_{hex}$) – Instanz 1...128 ... 294
Tabelle 33: Input Fieldbus Variable UDINT (AA $_{hex}$) – Common service ... 294
Tabelle 34: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Klasse ... 295
Tabelle 35: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Instanz 1...128 ... 295
Tabelle 36: Input Fieldbus Variable UDINT Offset (AB $_{hex}$) – Common service ... 295
Tabelle 37: Output Fieldbus Variable UDINT (AC $_{hex}$) – Klasse ... 296
Tabelle 38: Output Fieldbus Variable UDINT (AC $_{hex}$) – Instanz 1...128 ... 296
Tabelle 39: Output Fieldbus Variable UDINT (AC $_{hex}$) – Common service ... 296
Tabelle 40: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Klasse ... 297
Tabelle 41: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Instanz 1...128 297
Tabelle 42: Output Fieldbus Variable UDINT Offset (AD $_{hex}$) – Common service ... 297
Tabelle 1: 1-Kanal-Digitaleingangsklemmen mit Diagnose ... 300
Tabelle 2: 2-Kanal-Digitaleingangsklemmen ... 300
Tabelle 3: 2-Kanal-Digitaleingangsklemmen mit Diagnose ... 300
Tabelle 4: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten ... 301
Tabelle 5: 4-Kanal-Digitaleingangsklemmen ... 301
Tabelle 6: 8-Kanal-Digitaleingangsklemmen ... 301
Tabelle 7: 8-Kanal-Digitaleingangsklemme PTC mit Diagnose und Ausgangsdaten ... 302
Tabelle 9: 16-Kanal-Digitaleingangsklemmen ... 302
Tabelle 10: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten ... 303
Tabelle 11: 2-Kanal-Digitalausgangsklemmen ... 303
Tabelle 12: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten ... 304
Tabelle 13: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506 ... 304
Tabelle 14: 4-Kanal-Digitalausgangsklemmen ... 305

Tabelle 15: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 305
Tabelle 16: 8-Kanal-Digitalausgangsklemmen 305
Tabelle 17: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 306
Tabelle 18: 16-Kanal-Digitalausgangsklemmen 306
Tabelle 19: 8-Kanal-Digitalein-/ -ausgangsklemmen 307
Tabelle 1: 1-Kanal-Analogeingangsklemmen 308
Tabelle 2: 2-Kanal-Analogeingangsklemmen 308
Tabelle 3: 4-Kanal-Analogeingangsklemmen 309
Tabelle 3: 3-Phasen-Leistungsmessklemme 310
Tabelle 3: 8-Kanal-Analogeingangsklemmen 311
Tabelle 1: 2-Kanal-Analogausgangsklemmen 312
Tabelle 2: 4-Kanal-Analogausgangsklemmen 312
Tabelle 3: 8-Kanal-Analogausgangsklemmen 313
Tabelle 1: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003) 314
Tabelle 2: Zählerklemmen 750-404/000-005 315
Tabelle 3: Zählerklemmen 750-638, 753-638 315
Tabelle 4: Pulsweitenklemmen 750-511, /xxx-xxx 316
Tabelle 5: Serielle Schnittstellen mit alternativem Datenformat 316
Tabelle 6: Serielle Schnittstellen mit Standard-Datenformat 317
Tabelle 7: Datenaustauschklemmen 317
Tabelle 8: SSI-Geber Interface Busklemmen mit alternativem Datenformat 318
Tabelle 9: Weg- und Winkelmessung 750-631/000-004, --010, -011 318
Tabelle 10: Incremental-Encoder-Interface 750-634 319
Tabelle 11: Incremental-Encoder-Interface 750-637 319
Tabelle 12: Digitale Impuls Schnittstelle 750-635 320
Tabelle 13: Antriebssteuerung 750-636 320
Tabelle 14: Steppercontroller RS 422 / 24 V / 20 mA 750-670 321
Tabelle 15: RTC-Modul 750-640 322
Tabelle 16: DALI/DSI-Masterklemme 750-641 322
Tabelle 17: Übersicht über das Eingangsprozessabbild im „Easy-Modus“ 324
Tabelle 18: Übersicht über das Ausgangsprozessabbild im „Easy-Modus“ 324
Tabelle 19: Funkreceiver EnOcean 750-642 325
Tabelle 20: MP-Bus-Masterklemme 750-643 326
Tabelle 21: Bluetooth® RF-Transceiver 750-644 326
Tabelle 22: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645 327
Tabelle 1: KNX/EIB/TP1-Klemme 753-646 328
Tabelle 1: AS-Interface-Masterklemme 750-655 329
Tabelle 2: Systemklemmen mit Diagnose 750-610, -611 329
Tabelle 1: Binäre Platzhalterklemmen 750-622 (mit dem Verhalten einer 2 DI) 330
Tabelle 1: 1-Kanal-Digitaleingangsklemmen mit Diagnose 331
Tabelle 2: 2-Kanal-Digitaleingangsklemmen 331
Tabelle 3: 2-Kanal-Digitaleingangsklemmen mit Diagnose 332
Tabelle 4: 2-Kanal-Digitaleingangsklemmen mit Diagnose und Ausgangsdaten 332
Tabelle 5: 4-Kanal-Digitaleingangsklemmen 332
Tabelle 6: 8-Kanal-Digitaleingangsklemmen 333
Tabelle 7: 16-Kanal-Digitaleingangsklemmen 333

Tabelle 8: 1-Kanal-Digitalausgangsklemmen mit Eingangsdaten 334
Tabelle 9: 2-Kanal-Digitalausgangsklemmen 335
Tabelle 10: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 335
Tabelle 11: 2-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 75x-506 336
Tabelle 12: 4-Kanal-Digitalausgangsklemmen 336
Tabelle 13: 4-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 337
Tabelle 14: 8-Kanal-Digitalausgangsklemmen 337
Tabelle 15: 8-Kanal-Digitalausgangsklemmen mit Diagnose und Eingangsdaten 338
Tabelle 16: 16-Kanal-Digitalausgangsklemmen 338
Tabelle 17: 8-Kanal-Digitalein-/ -ausgangsklemmen 339
Tabelle 1: 1-Kanal-Analogeingangsklemmen 340
Tabelle 2: 2-Kanal-Analogeingangsklemmen 341
Tabelle 3: 4-Kanal-Analogeingangsklemmen 341
Tabelle 3: 4-Kanal-Analogeingangsklemmen für Widerstandssensoren [RTD] 341
Tabelle 3: 3-Phasen-Leistungsmessklemme 342
Tabelle 3: 8-Kanal-Analogeingangsklemmen 343
Tabelle 4: 2-Kanal-Analogausgangsklemmen 344
Tabelle 5: 4-Kanal-Analogausgangsklemmen 344
Tabelle 3: 8-Kanal-Analogausgangsklemmen 344
Tabelle 6: Zählerklemmen 750-404, (und alle Varianten außer /000-005), 753-404, (und Variante /000-003) 345
Tabelle 7: Zählerklemmen 750-404/000-005 346
Tabelle 8: Zählerklemmen 750-638, 753-638 347
Tabelle 9: Pulsweitenklemmen 750-511, /xxx-xxx 347
Tabelle 10: Serielle Schnittstellen mit alternativem Datenformat 348
Tabelle 11: Serielle Schnittstellen mit Standard Datenformat 348
Tabelle 12: Datenaustauschklemmen 750-654, /000-001 349
Tabelle 13: SSI-Geber Interface Busklemmen mit alternativem Datenformat 750-630, /000-0xx 349
Tabelle 14: SSI-Geber Interface Busklemmen mit alternativem Datenformat 750-630/000-00x 350
Tabelle 15: Inkremental Encoder Interface Busklemme 750-631/000-004, --010, -011 350
Tabelle 16: Inkremental Encoder Interface Busklemme 750-634 351
Tabelle 17: Inkremental Encoder Interface Busklemme 750-637 351
Tabelle 17: Digitale Impuls Schnittstelle 750-635 352
Tabelle 18: Antriebssteuerung 750-636 352
Tabelle 19: Steppercontroller RS 422 / 24 V / 20 mA 750-670 353
Tabelle 20: RTC-Modul 750-640 354
Tabelle 21: DALI/DSI-Masterklemme 750-641 355
Tabelle 22: Funkreceiver EnOcean 750-642 355
Tabelle 23: MP-Bus-Masterklemme 750-643 356
Tabelle 24: Bluetooth® RF-Transceiver 750-644 357
Tabelle 25: Schwingstärke/Wälzlagerüberwachung VIB I/O 750-645 357
Tabelle 26: AS-interface Masterklemme 750-655 358
Tabelle 1: Systemklemmen mit Diagnose 750-610, -611 359

Tabelle 2: Binäre Platzhalterklemmen 750-622 359
Tabelle 1: MODBUS-Tabelle und -Funktionscodes 362
Tabelle 1: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Busklemmen. 365
Tabelle 1: Beschreibung der Beispielbedruckung der ATEX- und IECEx-zugelassenen Ex i Busklemmen. 367
Tabelle 1: Beschreibung der Beispielbedruckung der Busklemmen gemäß NEC 500 369

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Telefon: 05 71/8 87 – 0
Telefax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com